TECHNISCHE UNIVERSITÄT MÜNCHEN

Frameworks for Large Scale Annotation of Proteins, Proteomes and Meta-proteomes

Guy Yachdav

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Uni- versität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr. Alexander Pretschner

Prüfer der Dissertation:

1. Univ.-Prof. Dr. Burkhard Rost
2. Prof. Dr. Yana Bromberg,
   The State University of New Jersey/USA

Die Dissertation wurde am 18.05.2015 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 04.06.2015 angenommen.

Contents

Abstract

According to UniProt, the largest public database of protein sequences, less than one percent of all known proteins directly map to experimental annotations. With the continuous exponential growth of sequences, the annotation gap – the difference between known sequences and sequences annotated – slows down the pace of scientific discoveries. Bioinformatics bridges this gap through the development of computational methods for protein function and structure prediction, relying on statistics, machine learning, data mining, and natural language processing.

PredictProtein is a software suite and-an online resource that integrates a battery of protein functional and structural prediction methods. Taking a protein sequence as input, the software generates over 30 prediction results. PredictProtein has had a wide impact on research and has been cited by over 1,200 manuscripts.

Given bioinformatics ubiquitous and critical role in post-genomic biological research, there is a need to ensure that the systems supporting this effort are stable, amenable for change and ready to meet the demands of modern biology. The work at hand, then, describes the technical and scientific solutions that improve the functionality of PredictProtein. These can be applied to the design of any other bioinformatics system. First, a framework that ensures the system adheres to a set of software engineering best practices is introduced. Then, methods for systematic software dissemination, results aggregation and their quick retrieval are discussed. Finally, visualization tools that make results easily communicable to biologists are reviewed.

As biology moves toward the study of complete sets of proteins in organisms (proteomes) as well as protein samples recovered directly from environmental sources (meta-proteomes), this work also focuses on extending PredictProtein to create a system for rapid and comprehensive analysis of large sets of proteins in a systematic way. The system prototype has been used to annotate, analyze and draw biological inferences from a published experimental meta-proteomics datasets.

0. Preamble

The thesis at hand constitutes a publication-based dissertation. The methodologies and results as presented here – in particular sections 2.1 and 2.2.2 – have been published in the following peer-reviewed articles. The following manuscripts have been appended to this dissertation. A description of each publication and my own contributions can be found in Appendix D.

- **Yachdav G**, Kajan L, Vicedo E, Steinegger M, Mirdita M, Angermuller C, Bohm A, Domke S, Ertl J, Mertes C, Reisinger E, Staniewski C, Rost B. *Cloud prediction of protein structure and function with PredictProtein for Debian*. Biomed Res Int. 2013;2013:398968.

The work reported in this manuscript is described in <u>sections 2.1.1 – 2.1.5</u>

- **Yachdav G**, Kloppmann E, Kajan L, Hecht M, Goldberg T, Hamp T, Honigschmid P, Schafferhans A, Roos M, Bernhofer M, Richter L, Ashkenazy H, Punta M, Schlessinger A, Bromberg Y, Schneider R, Vriend G, Sander C, Ben-Tal N, Rost B. *PredictProtein--an open resource for online prediction of protein structural and functional features*. Nucleic Acids Res. 2014;42(Web Server issue):W337-43.

The work reported in this manuscript is described in <u>sections 2.1.6 – 2.1.8</u>

- **Yachdav G**, Hecht M, Pasmanik-Chor M, Yeheskel A, Rost B. *HeatMapViewer: interactive display of 2D data in biology*. F1000Res. 2014;3:48.

The work reported in this manuscript is described in <u>section 2.2.2</u>

During the duration of the work described here I have also co-authored the following manuscripts.

- Goldberg T, Hecht M, Hamp T, Karl T, **Yachdav G**, Ahmed N, Altermann U, Angerer P, Ansorge S, Balasz K, Bernhofer M, Betz A, Cizmadija L, Do KT, Gerke J, Greil R, Joerdens V, Hastreiter M, Hembach K, Herzog M, Kalemanov M, Kluge M, Meier A, Nasir H, Neumaier U, Prade V, Reeb J, Sorokoumov A, Troshani I, Vorberg S, Waldraff S, Zierer J, Nielsen H, Rost B. *LocTree3 prediction of localization*. Nucleic Acids Res. 2014;42(Web Server issue):W350-5.

The work reported in this manusscript is mentioned in <u>section 2.1.7</u>

- Garcia L, **Yachdav G**, Martin MJ. *FeatureViewer, a BioJS component for visualization of position-based annotations in protein sequences*. F1000Res. 2014;3:47.

The work reported in this manusscript is mentioned in section 2.2.1

- Corpas M, Jimenez R, Carbon SJ, Garcia A, Garcia L, Goldberg T, Gomez J, Kalderimis A, Lewis SE, Mulvany I, Pawlik A, Rowland F, Salazar G, Schreiber F, Sillitoe I, Spooner WH, Thanki AS, Villaveces JM, **Yachdav G**, Hermjakob H. *BioJS: an open source standard for biological visualisation - its status in 2014*. F1000Res. 2014;3:55.

Conents from this manuscript were not used in this work.

- B Sokouti, F Rezvan, **G Yachdav**, S Dastmalchi. *GPCRTOP: A Novel G Protein-Coupled Receptor Topology Prediction Method Based on Hidden Markov Model Approach Using Viterbi Algorithm.* Current Bioinformatics 9 (4), 442-451

Conents from this manuscript were not used in this work.

A manuscript discussing the work described in section 2.2.3 has been drafted and is currently being reviewed by co-authors. The manuscript will be submitted to Bioinformatics by June 2015. Co-authors include: Guy Yachdav, Sebastian Wilzbach, David Dao, Robert Sheridan, Jim Procter, Ian Sillitoe, Susana Lewis, Burkhard Rost and Tatyana Goldberg.

The work described in section 2.3 will be written up and submitted for publication to the journal Bioinformatics. This will be done by the summer of 2015. The list of co-authors will include: Guy Yachdav, Diana Iaacob, Jonas Raedle, Yana Bromberg, Thomas Clavel, and Burkhard Rost.

1. Introduction

1.1. The sequence annotation gap

The following section introduces the data deluge challenge facing modern biology

The Protein Structure Initiative (PSI) project was a decade-long effort funded by the US National Institute of Health (NIH) in two phases. It encompassed up to fourteen US-based research centers and cost $270 million during the pilot phase and $325 million during the production phase. The goal of the project was to increase the number of solved protein structures as well as research and improve methodologies that will allow for faster, cheaper ways to elucidate protein structure in the future. The impetus for this large scale effort was motivated by the idea that increasing the number and breadth of the protein structural repertoire will support and accelerate research into better understanding the function of proteins, shed light on how altered structures can contribute to disease and help identify new targets for drug development (1). The PSI effort resulted in the deposition of an unprecedented, over-than 5900 newly and updated protein structures onto the Protein Data Bank (2) a central archive for protein structure.

The UniProt database (3), the largest publicly available sequence database, also logs the results of experiments that relate to the exploration of protein structure and function. UniProt is mainly composed of two sections: 1) UniProt/SwissProt - containing proteins with experimentally verified and reviewed annotations and 2) UniProt/Trembl - containing all collected sequences that lack expert annotations.

While the growth of UniProt/SwissProt is, in part, a result of efforts such as the PSI, a recent survey of UniProt reveals that UniProt/SwissProt and other expert annotation databases (such as the Protein Data Bank) are growing at a much slower pace than the growth rate of known protein sequences; This is not surprising given the way in which modern molecular biology has developed; advances in next generation sequencing and the steep decline in the cost of sequencing per base-pair (4-8) have resulted in an anomaly -- currently the exponential growth rate of UniProt/Trembl outpaces the growth rate of UniProt/SwissProt by 70-fold (Jan-Sep 2014 statistics (3)).

Comparing the number of entries in UniProt to those in PDB also shows a striking difference between the size of sequence space (the pool of all known sequences) and structure space size (the pool of all known structures); at the end of 2014 the PDB archived 105,426 protein structures whereas UniProt has logged over 82 million protein sequences (Sep 2014 statistics (3)).

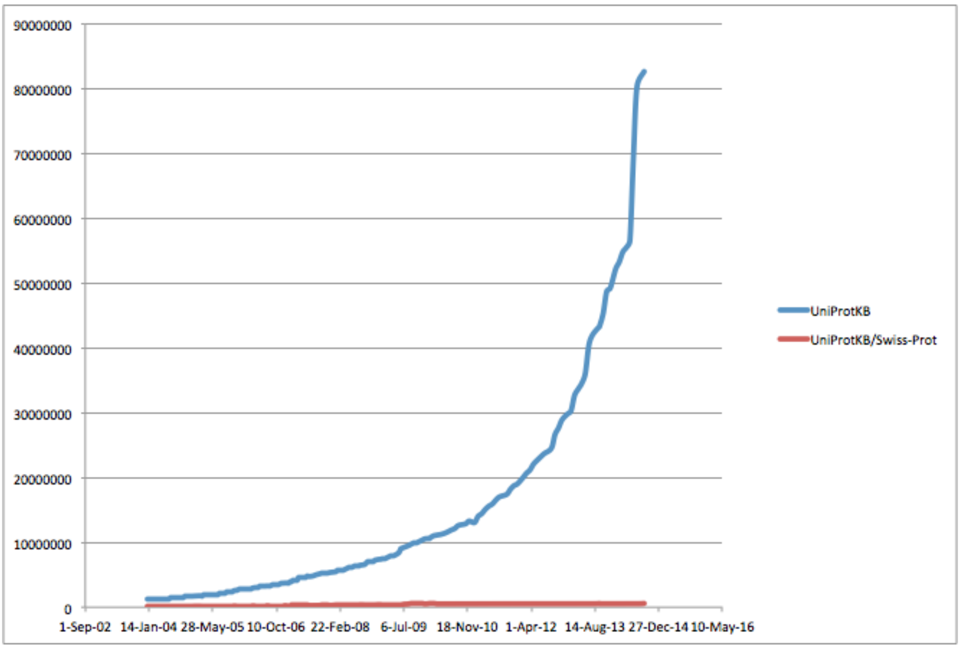


Figure 1 – growth of UniProt (blue line) as compared with the expert annotation UniProt/SwissProt section (red line). Statistics are for the period 1-Jan-2004 through 3-Sep-2014. All data compiled from (3).

The annotation gap is then defined as the difference ratio between the growth of the pool of known sequences and the set of experimentally verified annotated sequences. The annotation gap increases daily, driven by current technology available to researchers, that allows for greater opportunity of discovering new protein sequences at a lower cost than the process of generating experimental verified annotation.

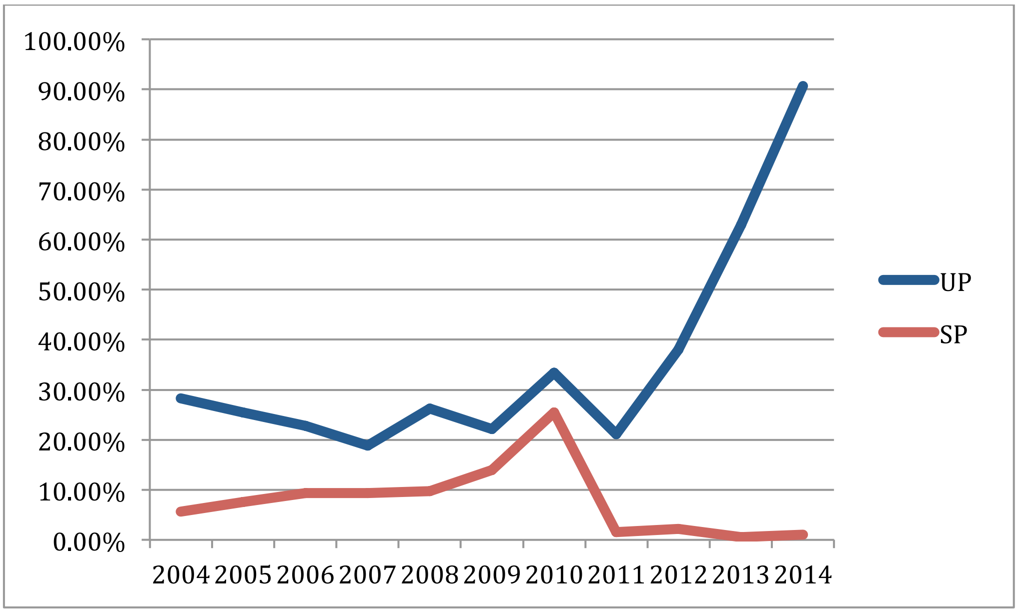


Figure 2 – the annotation gap exemplified by the difference in year over year growth rate of the UniProt(blue line) as compared with SwissProt(red line) for the period 2004-2014. Starting in 2011 the UniProt/SwissProt growth rate has plateaued around the 1.5% year while UniProt growth rate increased by an average of 63.4% each year.

By the end of 2014, the genomes of more than 2000 organisms had been completely sequenced (9). Many large-scale efforts have aimed at providing annotations for these sequences, for example, Gene Ontology (GO) (10), the Human Proteomics Initiative (HPI; http://www.uniprot.org/program/chordata/statistics) (11), and as mentioned above the Protein Structure Initiative (PSI) (1). GO, systematically describes biological function using ontologies that encompass molecular function, cellular localization and biological processes. However, not even the best ontologies can rely solely on expert annotations, because the experts are unable to keep up with the rapid influx of new data. From among the ~82 million proteins of known sequence to date, a bit over 30,000 proteins have been manually annotated with a GO term (www.ebi.ac.uk/GOA/uniprot_release.html).

1.2. In silico annotation of protein structure and function

The following section discusses the contribution of computational method in bringing the annotation gap introduced in the previous section

For the past two decades computational methods relying on statistics, machine learning, data mining, natural language processing as well as other strategies have been improving in accuracy returning annotations useful for verification of experimental results as well as elucidating scientific insight such as the prediction of protein function.

Indeed many methods that predict aspects of protein folding are now readily available for the community. The latest (2014) Critical Assessment of Techniques for Protein Structure Prediction 11 (CASP11) challenge lists 84 (12) prediction servers as participants. The prediction categories in the bi-annual CASP experiment represent current advancements and challenges in the field. For instance the prediction category Homology Modeling in which 37 groups participated during the CASP 6 (13) experiment (2004) was no longer featured during the CASP 11 experiment. Whereas the prediction of protein disordered regions have remained a challenge and in CASP 11 more groups participated in this category as compared to CASP6 (28 vs. 20). Similarly the Critical Assessment of Function Prediction (CAFA) also feature a host of prediction methods aimed at filling the protein function annotation gap (54 methods participated in the single CAFA assessment to date) by providing new strategies to predict aspects of protein function.

With the improvement and wider access to reliable predictions and at the same time the lack of availability of enough experimental data, biological databases now integrate predictions alongside data extracted from experimental resources. For instance STRING (14) is a database of protein-protein interaction that combines known experimentally verified interactions with predicted ones. Also, a growing number of prediction methods now collect annotations from prediction methods where no experimental data is available. In this strategy a method will implement a mechanism that will digest predictions into a single data models, and will consider the prediction error rate, alongside other statistical measures and summarize an overall reliability index for the resulting output. As an example, consider SNAP (15, 16) – a method for that predicts the impact of non-synonymous SNPs. SNAP uses predicted secondary structure, predicted solvent accessibility, and predicted flexibility as input features to its own neural network based prediction algorithm.

1.3. Combining annotation methods into meta-servers

The following section discusses the role of meta-server as an online central access point to a battery of computational methods

Traditionally, developers made their computational methods available to a small community of researchers. Such earlier packages required that the end user posses a high degree of technical competency. Earlier commercial packages combined these methods into a more user-friendly program suites such as the GCG (17) package. With the advent of the Internet and the appearance of web servers (Rost & Schneider, 1999) developers were making their resources available online (18). The overwhelming acceptance of the web as the primary resource for data retrieval created a new challenge of locating relevant and reliable resources. The BCM-Launcher (19) was one of the early solutions that offered a single portal allowing the access to a variety of tools. The single portal model, also known as the meta-server, simplifies the process of resource discovery by offering a central location for a set of contextually related tools. Furthermore the portal abstracts the technical need of constructing a workflow and handling a set of diverse and unstandardized file formats.

The growing need for higher degrees of collaboration across resources and data endpoints has brought on projects that aimed at building automatic and semi-automatic data discovery and data sharing infrastructures (20-23). While automation in data integration remains an important goal, still one aspect cannot be accomplished by any automation, namely the scientific combination of methods. For example, we know that signal peptides are often confused with membrane helices yet no method establishes a particular threshold that would allow an automatic distinction between the two classifications. Furthermore, the wealth of powerful tools and servers is, only utilized by a fraction of biologists who would be able to profit from them. Especially for non-experts it can be very time-consuming to find out which services exist, what they can or cannot do, how to use them and how to feed results from one service to the next in the right format. Therefore, biologists still turn to meta-servers as a primary point of access to computational resources (24).

Some of the core resources provided for the community are servers that group a set of tools and databases into a single meta-server. The Protein Data Bank (2) maintained by groups from Rutgers University and from the University of California, San Diego offers a Protein Feature View in which functional motifs, structured regions, disordered regions and Pfam domains (25) are all stacked into a single visualization. By using such view against regions where a 3D structure is available it is possible to see how a certain domain in a PDB entry relates to the full-length protein sequence.

Another example for a meta-server is PSIPRED (26, 27). The PSIPRED server is maintained by David Jones group at the University College London and provides protein access to a set of protein structure prediction and sequence analysis methods. PSIPRED main strength is in recognizing putative domains by combining secondary structure data searches for PDB chains hits, and domain boundaries prediction.

Finally, the MPI Bioinformatics Toolkit (24) provided by the Max Plank Institute for Developmental Biology in Tubingen is an interactive web service which offers access to a great variety of public and in-house bioinformatics tools. These are grouped into different sections that support sequence searches, multiple alignment, secondary and tertiary structure prediction and classification.

1.4. Sequence determines structure determines function

This section surveys the scientific paradigm that motivates the study of protein structure and function

Proteins are polymeric macromolecules involved in a vast array of functions within all living organisms (28). As enzymes, proteins play a critical role throughout the DNA replication machinery by assembling complementary nucleic acids to a template strand and synthesizing DNA. As ligand transporters, proteins bind to small molecules and carry them around the multicellular organism, or as membrane transporters, proteins assist in diffusing substances across the cell membrane. Proteins also play a crucial role in metabolism, facilitating enzymatic activity essential for the harvesting of energy (catabolism) and catalyzing the construction of new cell components (anabolism).

Preutz and Kendrew (29) provided an initial evidence for the relationship between a protein’s spatial conformation (fold) and its ability to carry out a certain function. By determining the 3D structure of the myoglobin protein and the Hemoglobin complex, Preutz and Kendrew demonstrated how the fold of the protein renders its ability to bind to oxygen molecules, transporting them from the respiratory system across the rest of the living organism.

The physical structure of a protein, it was shown, determines its ability to bind to other molecules, interact with them through chemical reaction and ultimately perform some activity. As an example, consider the Hexokinase enzyme that catalyzes energy production within the cell by binding glucose and ATP. The interaction between the enzyme and its substrates is only possible through a set of binding sites – i.e. a set of regions on the proteins surface that can create weak non-covalent bonds with the interacting molecule. Binding sites can be described as cavities on the protein surface (figure 1) and are characterized by a physical shape and chemical composition that is highly specific to the surface contours of the molecule they bind (the ligand).

Christian Anfinsen's Dogma (30) postulates that the native structure of a protein is largely determined by its amino-acid sequence. This dogma induced the sequence→structure→function paradigm (sequence determines structure determines function) suggesting that largely the protein sequence determines the way the protein folds and that this fold (3D structure) in turn determines the protein's function.

In recent years some researchers saw reason to slightly alter the paradigm with the discovery of protein disorder. Disordered proteins lack a stable unique ordered three-dimensional structure. Still, the lack of a defined structure appears to be the key for a variety of important biological functions that involve cell cycle control, gene regulation or signaling (31, 32). Nonetheless the sequence to structure to function paradigm still holds and remains one of molecular biology's cornerstones.

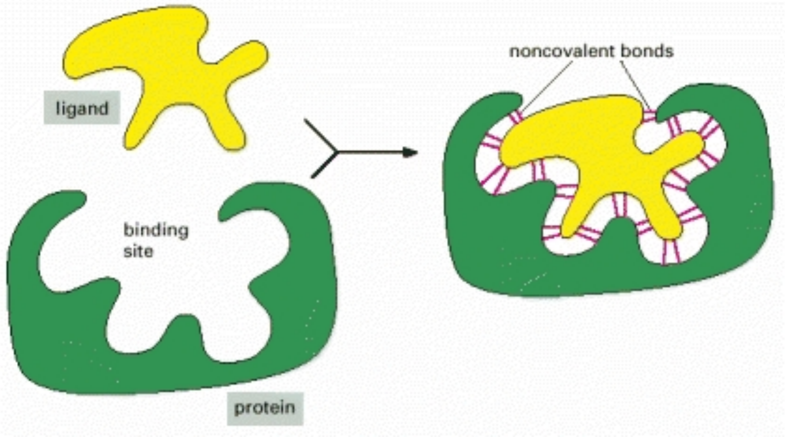


Figure 3 - many weak non-covalent bonds are needed to bind tightly to a second molecule (called a ligand) A ligand must therefore fit precisely into a protein's binding site, like a hand into a glove, so that a large number of non-covalent bonds can be formed between the protein and the ligand (adapted from (28))

1.5. Thesis objectives

This section details the goals pursued in the work described in this thesis

Aim 1 - Create a reliable and maintainable protein annotation server.

PredictProtein (PP) is a protein sequence analysis software suite that has been developed in our group over 23 years (went online in 1992). PP's unique approach combines many analysis and prediction methods into a "meta-server" which lends itself to comprehensive protein sequence analysis prior or in parallel to the experimental discovery process. In realizing this aim I have been laying the groundwork required to make PP a reliable and scalable system for the analysis of individual proteins as well as to large datasets. All software components that make up the PP server have been reviewed and standardized according to software development best practices. The software that orchestrates the pipeline activity have been rewritten and made flexible and fault-tolerant. A new repository that indexes, stores and retrieves PredictProtein results has been constructed to enable rapid access to PP results. Finally, a modern data-driven web application has been developed to allow efficient and reliable access for the community of 10,000 PP users that access the resources each month. The web server incorporates a data visualization application that makes use of the visualization tools developed as part of aim 2 in this work.

Aim 2 - Develop visualization tools required for the presentation of protein annotation data on the web. Big-data becomes increasingly important in life science as data grows rapidly in volume and complexity. Existing tools aim at reducing the complexity level of arbitrarily large textual data (often organized in tabular format) into more intuitively accessible visual representation. As more data is consumed over the web by the mediation of web browser applications there is a need to create software tools that are native to the web browser, i.e. that do not require additional, complicated deployment procedures. I developed a set of tools necessary to visualize the data annotations generated by the PP pipeline. All tools were developed in JavaScript, a language native to modern web browsers and ubiquitously employed across the web to deliver interactive images. The tools were incorporated into the PP web application as well as made available to the community as stand alone viewers through the BioJS (33) visualization library and registry.

Aim 3 - Establish a pipeline for the analysis of proteomes and meta-proteomes.

Proteomics and meta-proteomics experiments aim at studying complete organisms or random communities of microorganisms extracted from cultured or environmental samples. Computational methods have been developed to help in the characterization of proteins sequenced in those studies. Yet there is still a need for systems that would assist in the comparative analysis of datasets captured from different samples. Such systems are especially important in meta-proteomics studies where multiple organisms are present and often composed of unknown or sparsely characterized proteins. In this part of the project the PP software was extended to provide rapid annotation for datasets of proteins. The system enables the functional annotation and analysis of large genes/proteins lists. The prototype version of the system is designed for the needs of meta-proteomics studies of the gut bacteria. The system can be used to A) provide additional evidence or challenge existing findings, and to B) highlight new 'interesting' proteins given a set of annotations that will support follow on research.

2. Methodology and results

2.1 The PredictProtein meta-server

2.1.1 Background

The following section introduces the PredictProtein server and the guidelines used in rebuilding a robust bioinformatics annotation system

The PredictProtein (PP) server is a protein structure and function annotation service that went online at the EBML Heidelberg in 1992. The service started as one of the first online resources for the prediction of secondary structure and transmembrane helices (34, 35). From 1999 to 2009, the server operated from Columbia University (New York, NY) and in 2009 it moved to the TUM (Munich, Germany). PP was one of the first services realizing state-of-the-art protein sequence analysis, and the prediction of structural and functional features in a single server. The primary goal of the server has always been to provide a system optimized to meet the demands of experimentalists not highly experienced in bioinformatics. To this end, the PredictProtein pipeline has grew to integrate over 30 prediction methods, database search tools, post processing and parsers as well as supporting libraries. Furthermore results are presented as both text and a series of intuitive, interactive and visually appealing figures.

A combined software suite is an agglomeration of software pieces, each created to fit a different set of technical specifications, each often written by different developers and coded in various programming languages. On top, suites often depend on other programs and libraries and require distinct input formats. The issue of software dependencies is most pronounced since during software's life span updates and changes to underlying code, and libraries are introduced. Such modifications can degrade capability and even render the software tool unusable. Furthermore, maintainability of the software suite becomes an even bigger challenge as the system grows in scale and scope and changes to the software tools integrated are introduced more frequently.

To explain the complexity in designing and maintaining a system such as PredictProtein consider that the current version of the system incorporates over 30 computational methods. Each such method relies on a set of parsers, compilers, modules and external tools so it could properly function. Each of those dependencies in turn relies on another set of system utilities and system libraries. Figure 4 shows a selected set of tools that were incorporated into the system, yet for the sake of clarity this list is restricted only to the main bioinformatics packages that have been integrated. The list does not show the complete dependency graph of the system which includes all supporting software tools. The rostlab packages wiki page[1] lists 89 software packages that have been integrated into PredictProtein. Figure S2 (Appendix C) shows a dependency graph for the

[1] https://rostlab.org/owiki/index.php/Packages#Package_overview

software that is integrated and supports PredictProtein and their dependency relations.

Furthermore, the tools integrated by PredictProtein and maintained by our group were written by over 50 developers (a non-exhaustive-list of main contributors can be found on the PredictProtein website (36)) over the course of 23 years. The code base includes code written in the following programming languages: Python, PERL, Java, FORTRAN, R, C/C++ and the following Unix shell scripting environments and utilities: tcsh, BASH, make, awk.

The challenge of maintaining software that explicitly depends on dozens of tools and implicitly on hundreds of dependencies is compounded by the time element and the Information Technology reality that is driven by constantly changing computing environment. A third (12 methods) of the tools integrated into PredictProtein were incorporated over the past five years.

As the PredictProtein system has grown in size and scope and as it became more mature, the need to implement a set of guidelines and best practices (37) that will guarantee the continued scope expansion while ensuring the stability of the system arose. Following is a set of principles adopted and implemented throughout the redevelopment of the PredictProtein system:

- **Resilient:** The system stays responsive in the face of failure. Failures are contained within each component, isolating components from each other and thereby ensuring that parts of the system can fail and recover without compromising the system as a whole. The client of a component is not burdened with handling its failures.
- **Flexibility:** The system stays responsive under varying workload. Systems can react to changes in the input rate by increasing or decreasing the resources allocated to service these inputs. This implies designs that have no contention points or central bottlenecks.
- **Responsiveness** - the system responds in a timely manner. Responsiveness enhances usability and utility. Responsive systems focus on providing rapid and consistent response times, establishing reliable upper bounds so they deliver a consistent quality of service. This consistent behavior builds end-user confidence, and encourages further interaction.

2.1.2 Improvements to source code

The following sections (2.1.2-2.1.6) will describe the steps taken in this project to create a more resilient, flexible and responsive system.

As a first step for source code maintenance we collected all source for component methods and deposited those into a version control management system. Version control management systems keep track of changes introduced to the code base throughout the development project life cycle, enable documentation of those changes through logging, and provide a central repository that multiple developers can access, share code and collaborate. Source code deposited into version control was “cleaned up” from convenience copies, a software development anti-pattern in which the code base repository is being cluttered with multiple backup copies of the same source files.

Deposited source was also reviewed and “cleaned up” from dysfunctional code (code that was not being called during the execution time of the program or did not serve any functional purpose). Excessively commented out code and nonsensical comments were also removed to reduce source code obfuscation. A naming scheme that enforces unified, meaningful naming of variables was enforced. Runtime issues that raised system warnings were resolved and nonsensical warnings generated by the program were muted.

The use of “exotic” modules and libraries has been disallowed. Exotic modules are ad-hoc software libraries that haven’t been sanctioned by and official release process. Usually such modules can be obtained from online sources but are rarely backed by a community of maintainers. Due to their uncertain maintenance state, exotic modules included into software can burden the development team with bug fixes and keeping the code current.

To make sure that errors were contained within components, fault tolerance was addressed in three levels:

1. User level errors were handled through user input validation
2. System level run time errors were captured and reported to the error stream
3. Exit code are used to report process state to any downstream process.

2.1.3 Software packages

Addressing the resilience guideline we have delineated the different bioinformatics tools that are incorporated into the system. Each of those tools is defined as a component. Roughly speaking a component could be a complex software, a set of algorithms, a database or a simple script; each component is usually composed of specialized code that performs a specific operation and relies on a set of other tools as well as specific programming language libraries, modules and plugins defined as dependencies. By identifying the set of components and dependencies it is easier to map the scope of operation within the overall system, ie points of failures can be easily traced and isolated at the component level.

Once the components-dependency maps are charted, we turned to organize those maps into software packages. Software packages include the source code of the software as well as a set of instructions that inform the operating system how to automatically maintain the software during deployment, upgrade and. Technical details such as software version, dependency packages as well non-technical details such as the name(s) of the developer(s) and maintainer(s) and a free text description of the software capabilities are all given within the package. The instructions contained in the software package are used by a *package manager* utility.

The package manager utility automatically puts together an installation plan -- a tree-like graph that resolves all dependencies, pulls in necessary versions, highlights version conflicts, verifies necessary storage space and prioritizes the installation steps. The package manager ensures an organized, smooth and error free deployment process that maintains the integrity and stability of both the installed software and the host operating system. The build and installation process of a package on the target system makes use of auto-tools. Auto-tools are a set of Unix based utilities that streamline and automate the software deployment process.

PredictProtein method components were packaged using auto-tools. During the package bundling step each package is clearly and automatically labeled with the software name and its serial version number. Software packages were then adapted to work with the Debian Linux operating system. The Debian operating system is a widely recognized and accepted Unix-like operating system that is composed entirely of open-source software, most of which is under the GNU General Public License, and packaged by a group of individuals known as the Debian project. Within the Debian project, the Debian-med group aims to develop Debian into an operating system that is particularly well fit for the requirements for medical practice and biomedical research. The Debian-med project maintains a software and databases repository that primarily targets bioinformatics. Making PredictProtein components available through the Debian repository resulted in one of the largest ever contribution to the Debian-med repository in which 89 software tools and libraries (38) were added to the repository.

Beyond the sizable contribution of open source software to the Debian med repository, the packaging effort also had two immediate benefits that address the needs for open computing in science (39):

(ii) Transparency - the packaging process ensures that the source code for the packaged software remains open and in the public domain for other developers to be able to build upon and extend current work as well as provide the necessary transparency required for end users who wish to inspect the way in which results were derived.

(ii) Replicability (capable of replication) - packaging software and publishing it through a software repository aims at making sure that the software would be deployed to all system in a similar manner, enforcing that similar versions should perform consistently across similarly configured (as well as different) systems.

2.1.4 Virtual machine

Using a software repository to selectively pick and install different packages can be used deploy the PredictProtein suite onto a host Debian Linux system. However an alternative software distribution method that allows a turn-key solution to be quickly deployed into an end system or a cloud based service was implemented as the PredictProtein Machine Image (PPMI), a disk image that can be run in a virtualized environment on a host system. This image contains a minimal installation of Debian with the command line version of PredictProtein. Databases are provided as a separate disk image. The PPMI is bootable on server instances in cloud infrastructure services, or on locally installed virtualization software. The latter allows for a cross-platform solution to use PredictProtein. After booting the machine image, a friendly message at the login prompt offers usage tips and directions to documentation. A “Getting Started with PredictProtein” guide is available online[2]. The PPMI and the data image are updated regularly and are freely available through the Rostlab website[3].

2.1.5 Flexible workflow construction

To address the flexibility guideline, we put together all components to a specialized, single PredictProtein workflow by utilizing the Make utility. Make is a utility that automatically builds executable programs and libraries from source code by reading files called makefiles which specify how to derive the target program. The makefile instruction file specifies targets to be made and a “recipe” of how to make them. Each prediction file was marked up as a target to be built (or made in the Make terminology) and the recipes included method execution specifications (e.g. the set of parameters that were calibrated to optimally run the method). For each target a set of required targets are also specified; when make encounter such specification it is required that the required target be made first before completing the current target. Since the make process lays out the process of building files ahead of execution and also checks for existing targets, it eliminated the need for redundancy in the build process, as existing targets are not built twice. Furthermore the make process also provides the ability to parallelize work, allow the build of two or more targets to happen simultaneously if those are not dependent on each other, thus utilizing resources most efficiently. The make process therefore provides the core process that orchestrate the execution of over 30 different bioinformatics packages and generate the resulting annotations.

[2] https://wiki.debian.org/DebianMed/PredictProtein

[3] http://rostlab.org/services/ppmi/

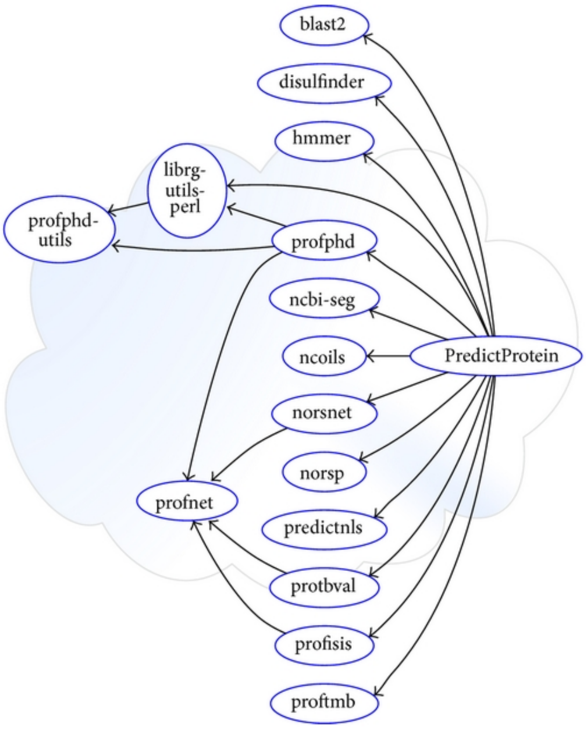


Figure 4 - package dependencies for PredictProtein. Arrows represent "depends on" relationships. Only significant dependencies are shown for clarity. "profnet" is a FORTRAN based neural network component used by multiple PP core components. Convenience copies of "profnet" for the prediction methods "profphd," "norsnet," "profbval," and "profisis" have been merged to a single "profnet" package. Similar merging was done for all code convenience copies.

2.1.6 Responsiveness using PredictProtein cache

The PredictProtein Cache (PPcache) is a database that currently holds pre-calculated results for over 13 million unique proteins, including all proteins of model organisms. The PredictProtein web server makes use of the PP cache by retrieving results directly from the cache. For results older than three month, users are given the option to re-run the query, thereby updating the PPcache. If no result exists in the PPcache, the job is processed, and users are notified upon job completion. PPcache currently requires roughly 100TB of disk space.

2.1.7 Methods included

The following section lists some of the core methods and databases included in the PP suite and server

Prediction methods - a prediction method is a single or set of specialized algorithms leveraging statistical model to forecast a particular feature related to the protein structure or function. For instance the profisis prediction method (40) forecasts the likelihood of each residue in a given amino acid sequence could be involved in protein-protein binding. Following is a list of highlighted prediction methods that were integrated into PredictProtein and their performance evaluations.

TMSEG transmembrane helix predictions - TMSEG (41) predicts alpha-helical transmembrane proteins, the position of transmembrane helices, and membrane topology. The method uses a novel segment-based neural network to refine the final prediction. TMSEG was developed and evaluated on 166 transmembrane proteins extracted from PDBTM (42) and OPM (43), and on 1441 proteins from the SignalP4.1 dataset (44). In our hands, TMSEG appears to complement and improve over the best existing methods (e.g. PolyPhobius (45) and Memsat3 (26)) predicting all membrane helices correctly for about 60% of all proteins. The method correctly identifies 98% of all transmembrane proteins with a false positive rate of less than 2%.

SNAP2: predict effect of mutations upon function - SNAP2 predicts the effect of single amino acid substitutions on protein function (46) . It improves over its predecessor SNAP (15) by using additional coarse-grained features that better classify samples with unclear evidence. With a two-state accuracy of 83% and an AUC of 0.91, SNAP2 performs on par or better than other state-of-the-art methods on human variants while significantly outperforming these methods for other organisms. For each protein we also predict the entire protein mutability landscape (47, 48), *i.e.* the functional effect of all possible point mutations. The results are displayed in a heat-map representation (49) of functional effects (Fig. 6C).

LocTree3 subcellular localization for all domains of life - LocTree3 predicts subcellular localization for proteins in all domains of life (50). The method predicts the localization in 18 classes (8 classes for transmembrane and 10 classes for soluble proteins) for eukaryotes, in six for bacteria and in three for archaea. LocTree3 successfully combines de novo (51) and homology-based predictions (52), reaching 18-state prediction accuracy over 80% for eukaryotes and a six-state accuracy over 89% for bacteria. The high level of performance and the large number of predicted classes make LocTree3 the most comprehensive and most accurate tool for subcellular localization prediction.

***metastudent* infers GO terms by homology -** the method metastudent (53) predicts Gene Ontology (GO) (10) terms through homology inference. It first BLASTs queries against proteins with experimental GO annotations taken from Swiss-Prot (54). Then, three algorithms independently choose which GO-terms to inherit. These differ in the amount and quality of alignment hits considered and how they assign a probability to each GO term. A meta-classifier combines the three through linear regression. metastudent achieves a maximum F1 score of 0.36 in the biological process ontology and of 0.48 in the molecular function ontology (53). Although this is slightly worse (within the error estimates (55)) than the best method for predicting GO terms (56), the advantage is that metastudent predictions can easily be traced back to the experimental annotations upon which they are based.

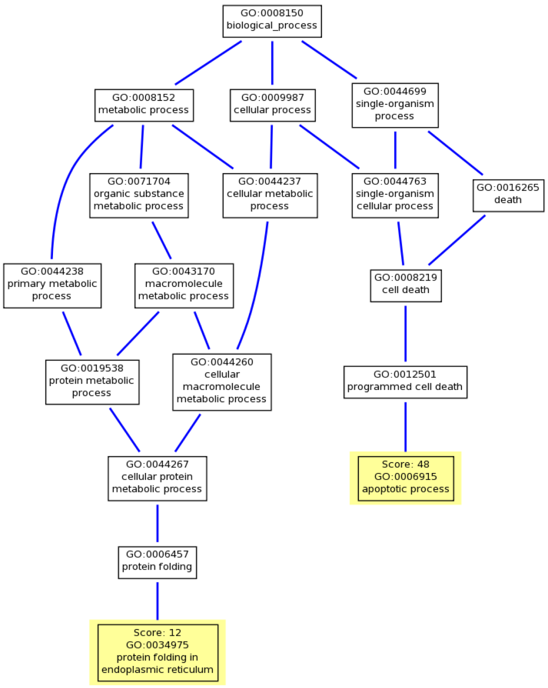


Figure 5 - metastudent predicts GO terms. metastudent (53) is a program in the PredictProtein suite that predicts the function of proteins in terms of Gene Ontology (GO) numbers (shown in each box are unique GO numbers, the Score for the prediction ranging from 0 (low) to 100 (high). The GO numbers are given along with a biochemical description, e.g. the most detailed prediction in shown in the yellow boxes at the bottom imply that the protein for which PredictProtein was run is involved in “protein folding in the endoplasmic reticulum” (which happens to be GO number 0034975) with a score of 12. The protein is also predicted to be part of the apoptotic process (cell death) with an even higher score of 48.

Meta-Disorder prediction of protein disorder - intrinsically disordered or unstructured regions in proteins do not fold into well-defined three-dimensional structures but may become structured upon binding to a substrate. Because of the heterogeneity of disordered regions, we have developed several methods predicting different types of disorder. UCON (57) combines protein-specific pairwise contacts predicted by PROFcon (58) with pairwise statistical potentials to predict long disordered regions that are rendered intrinsically unstructured by few internal connections. NORSnet (59) predicts disordered regions with NO Regular Secondary structure (NORS (60), i.e. long loops), separating very long disordered loops predicted by NORSp (61) from all other regions in the PDB (62). PROFbval (63, 64), trained on B-values in X-ray structures, predicts flexible residues in short disordered regions. Meta-Disorder (65) is a neural-network based meta-predictor that uses different sources of information, including the orthogonal disorder predictors mentioned above and others, e.g. IUPred (66) and DISOPRED (67). Meta-Disorder significantly outperforms its constituents (65, 68). A comprehensive, independent study (68), on disordered regions from the PDB and DisProt (69), suggested Meta-Disorder to be one of the top two methods available.

Protein-protein binding sites - residues that can bind other proteins are now predicted by ISIS2 instead of ISIS (40). ISIS splits a query sequence into windows of nine consecutive residues, encoding each window as a vector of features (e.g. PSI-BLAST amino acid conservation frequencies or predicted secondary structure). A neural network, trained on existing protein-protein binding residue annotations, determines whether a query residue can bind other proteins. ISIS2 has been trained on a large data set of PDB-annotated binding sites (70). A faster neural network implementation (70) and new methods for predicting residue features further improve the accuracy of ISIS2.

Protein-DNA, protein-RNA binding sites - protein-polynucleotide binding underlies important processes such as replication and transcription. SomeNA (71) predicts protein-polynucleotide binding on three levels. First, it predicts which proteins bind nucleotides. Second, it predicts the type of binding (RNA or DNA or both). Third, it predicts the protein residues that bind DNA or RNA. The first step is performed best: 77% of the proteins are correctly predicted to bind DNA and RNA. The distinction between the type of nucleotide is slightly more difficult: 74% of the proteins predicted to bind DNA and 72% of the proteins predicted to bind RNA were correct. Slightly over 53% of the residues binding DNA and/or RNA were correctly predicted. These levels of performance are at least 3-fold higher than random.

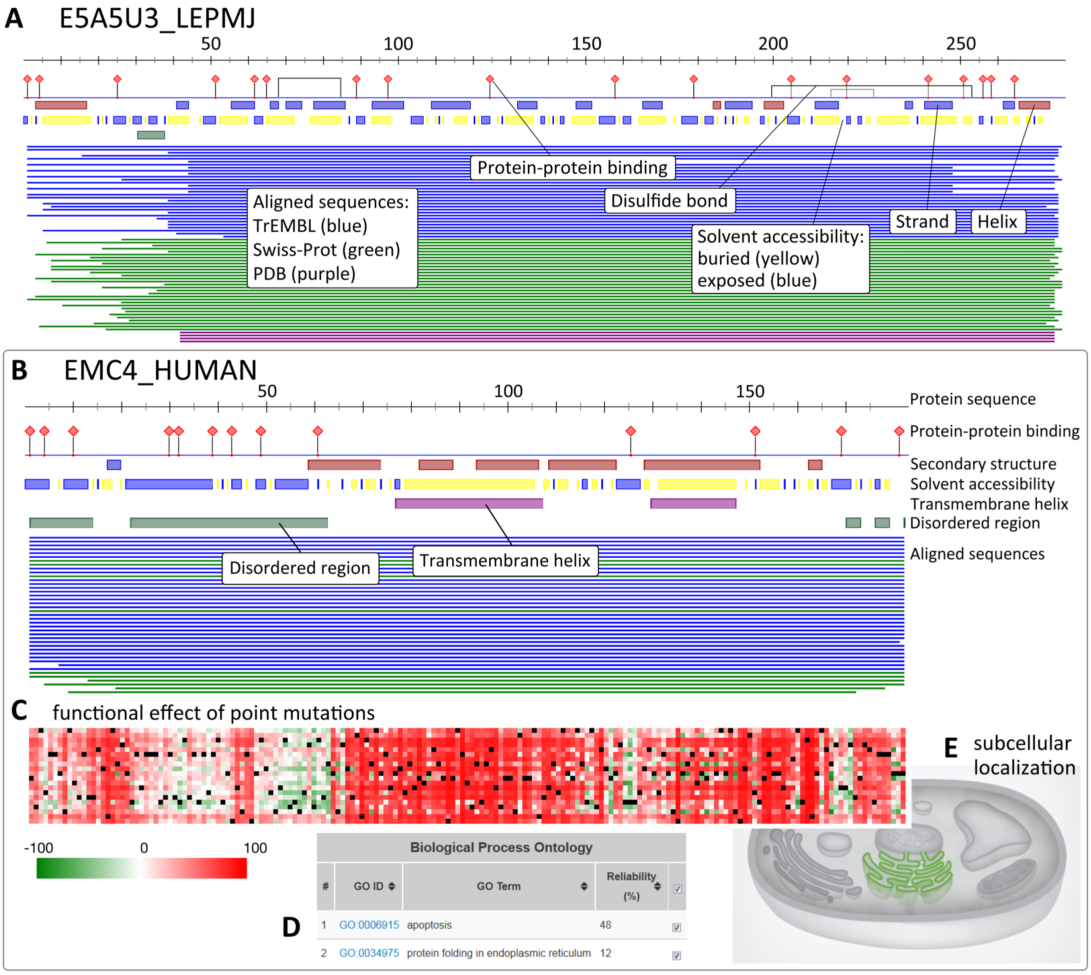
ConSurf conservation of surfaces explains function - ConSurf (72, 73) estimates the evolutionary rate in protein families. These rates are useful for protein structure and function prediction because they reflect constrains imposed on the general evolutionary drift (35, 74, 75). Queried with a protein sequence, ConSurf first finds related sequences in UniProt. Evolutionary rates of amino acids are estimated based on evolutionary relatedness between the protein and its homologues using either empirical Bayesian (76) or maximum likelihood (77) methods. The strength of these methods is that they rely on the phylogeny of the sequences and thus can accurately distinguish between conservation due to short evolutionary time and or conservation resulting from importance for maintaining protein foldability and function. If a structure is available, ConSurf maps the patterns of conservation upon the 3D structure. These patterns reveal crucial details about protein function.

Database Searches - a database search algorithm is optimized to perform a search and retrieval operation over a biological dataset. For instance the hmmsearch tool (25) takes a protein sequence as input and matches this input against a database of hidden Markov models (HMMs) based upon protein families. In PP sequences similar to the query are identified by standard, pairwise BLAST (78) and iterated PSI-BLAST (52) searches (75, 79) against non-redundant combination of PDB (2), Swiss-Prot (54) and TrEMBL (3). In addition, functional motifs are taken from PROSITE (80) and domains from Pfam (25).

2.1.8 The PredictProtein web server

The PredictProtein web server provides a central, online interface for the interaction with the PredictProtein annotation pipeline described in section 2.1. The web server is available at http://www.predictprotein.org presents a simplified, streamlined interface in which users can upload an amino-acid sequence on the server homepage thus creating a PredictProtein job (PP job). Every step in the life cycle of a PP job (whether received, in process, completed or in error mode) is being communicated back to the user with actionable options when available. Upon job creation the web server schedules the processing of the uploaded input sequences by the PredictProtein annotation pipeline on a 600 cores Linux cluster. Account registration is available for users who wish to create a repository of processed jobs to which they can refer to a later time. Users who prefer not to register for the service can use the open version of the service available at http://ppopen.rostlab.org.

Upon the normal completion of job processing, results become available through an interface that integrates a set of visual components (described in detail on section 2.2) and provides a navigation pane that allows users to easily browse different results classes. Figure 6 shows a sample of the results presentation available on the PredictProtein online interface.



Biological Process Ontology

#	GO ID	GO Term	Reliability (%)	☑
1	GO:0006915	apoptosis	48	☑
2	GO:0034975	protein folding in endoplasmic reticulum	12	☑

Figure 6 - visual results from PredictProtein (PP). The PP Dashboard Viewer shows a schematic of all position-based predictions and sequence alignments. A: Putative protein (UniProt AC E5A5U3). B: ER membrane protein complex subunit 4 (EMC4, UniProt AC Q5J8M3). The protein sequence is represented by a scale on top of the predicted features. Features presented include protein-protein binding sites (ISIS2), disulfide bonds (DISULFIND), structural features such as secondary structure state and solvent accessibility (PROFphd), transmembrane helices (TMSEG) and disordered regions (MD). Proteins aligned by PSI-BLAST (52) are shown as thin lines colored by database origin (PDB (2), Swiss-Prot (54) and TrEMBL (3)). Clicking on each line links to the database entry of the hit. For all elements, tooltips disclose the annotated feature, its position in the sequence and its type (prediction vs. database search). C: A complete analysis of the functional effect of point mutations on EMC4 shown in a heatmap (SNAP2). D: Predicted GO terms (metastudent) for EMC4 in tabular format. E: The predicted cellular compartment, ER membrane, for EMC4 (LocTree3) is highlighted in green in a schematic of a eukaryotic cell.

2.2 Data Visualization

The following section (2.2) reviews the motivation and work to create a set of visualization tools

Web pages are ideal tools for the dissemination of results and data. Dynamic interactivity is crucial in the discovery process, particularly for data-rich applications, as is the case of many websites that provide interfaces to biological databases. Databases storing genomic and other types of data have proliferated in the biological sciences, making them a data-rich, data-intensive set of disciplines. The visualization of these data plays a crucial role in their interpretation as it permits the ability to hide or to focus on a particular detail, enabling the researcher to shed light on specific hypotheses or to create new ones based on observed patterns. The sheer complexity of biological data, however, requires more complex technologies than the usual static pages when accessing them. They require dynamic visualization tools to allow real-time interactions and the usability of Web 2.0-based technologies. With the ubiquitous adoption of JavaScript as the de-facto technology to deliver rich, interactive and scalable graphics on the web by all major web-client software it is now possible to present the data generated by the methods integrated into PredictProtein in an accessible and intuitive way that will help researchers examine the results generated by multiple methods.

2.2.1 Feature viewer

Position-based annotation is one of the cornerstones of bioinformatics. A great number of databases, analysis and prediction methods are geared towards providing data mapped to specific sequence coordinates. In the case of proteins, the Pfam database (25) identifies, marks-up, and characterizes different functional regions within a given protein. The coordinates of these domains are often given in terms of the start and end position within the protein. UniProt (3) contains position-based annotations for structural regions, modified residues, and functional sites among others.

The Protein FeatureViewer is a JavaScript based component that lays out, maps, orients, and renders position-based annotations for protein sequences. This component is highly flexible and customizable, allowing the presentation of annotations by rows, all centered, or distributed in non-overlapping tracks. It uses either lines or shapes for sites and rectangles for regions. The result is a powerful visualization tool that can be easily integrated into web applications as well as documents as it provides an export-to-image functionality.

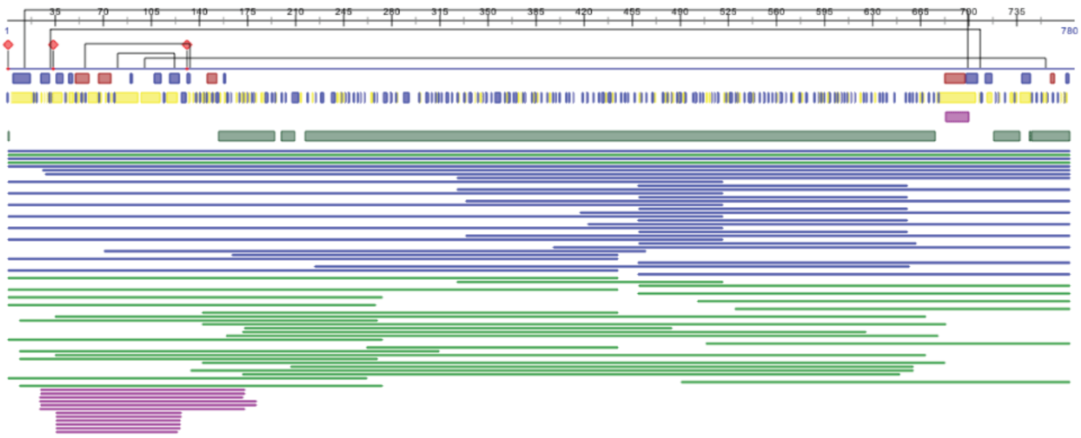


Figure 7 – the FeatureViewer component was adapted to show all position-based predictions generated by PredictProtein into a single interactive image. Position in the amino acid sequence can be oriented by using the provided scale (top of figure). The amino acid itself is represented by a blue line. Predicted sites (such as allosteric sites, protein-protein binding site, protein-DNA/RNA binding sites are represented by sticks and diamonds coming. The brackets connecting positions on the sequence represent intra-connected sites such as disulphide bonds. The boxes under the blue line represent various annotation where each annotation is typically laid out in different rows (called tracks). Refer to figure 6A and 6B for concrete example and labeling of the various annotations. Thin blue, green and purple lines represent aligned proteins found in TrEMBL, Swiss-Prot and PDB respectively.

2.2.2 Heatmap viewer

Biological data are often organized into matrices in which the rows signify different items of interest (a gene, a subject, a probe or a position in a sequence), while the columns describe different experiments, variations, or samples. Matrices are easy to process by algorithms. In contrast, the details in large matrices are often, at best, challenging for experts who want to "understand" the data. The information in matrices is usually better digested if presented by 3D plots or heat maps. Heat maps are essentially simplified versions of 3D plots that replace the 3rd dimension with color gradients, thereby conveniently displaying the information contained in matrices. Such heat maps allow for easy visual differentiation between high and low values in a matrix.

Such heat maps are, for example, commonly used to display microarray data as they quickly show which genes (rows) are differentially expressed under some conditions (columns). Microarray technologies utilize arrays of probes located on different exons for each gene and can be helpful in determining gene function by measuring transcription and translation levels under certain experimental conditions. The expression values for the differential expression may be presented at the exon level, correlated with protein domains, and may help to decipher a complex gene expression pattern.

The HeatMapViewer is a JavaScript-based component that lays-out and renders two-dimensional (2D) plots or heat maps that are ideally suited to visualize matrix formatted data in biology such as for the display of microarray experiments or the outcome of mutational studies and the study of SNP-like sequence variants. It can be easily integrated into documents and provides a powerful, interactive way to visualize heat maps in web applications. The software uses a scalable graphics technology that adapts the visualization component to any required resolution, a useful feature for a presentation with many different data-points. The component can be applied to present various biological data types. Here, we present two such cases – showing gene expression data and visualizing mutability landscape analysis.

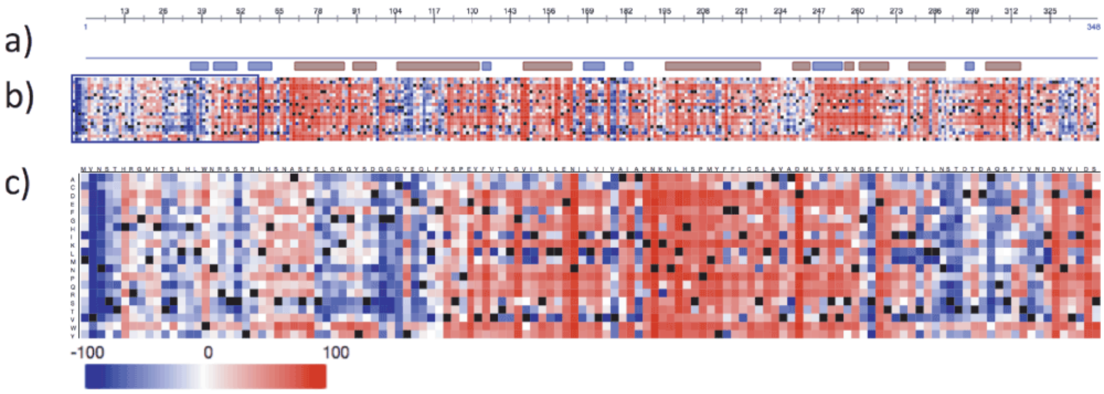


Figure 8 – The HeatMap Viewer component intergrated with FeatureViewer component (section 2.2.1). Panel a) sketches the secondary structure (helices in red, beta strands in blue). Panel b) shows the predictions of effects for each amino acid substitution. Effects are depicted as color intensities ranging from dark blue (high probability of no or little effect) over white (effect can not be predicted or only with very low reliability) to dark red (high probability of strong effects). Black depicts wildtype residues. The blue box marks the zoomed-in region shown in panel c). The *HeatMapViewer* provides a fast and easy way to represent high dimensional data in a visually comprehensible way that immediately conveys where mutations are likely to be deleterious. Mutability landscape studies (48) involve predicting the effect of all possible nsSNPs through computational methods, visualizing the predictions in heat maps and cross-linking these predictions with additional sources of information (such as secondary structure, active sites and correlated mutational behavior). To this end, heat maps (panels b, c) can easily distinguish between low effect regions (represented in blue) and high effect regions (represented in red) while additional information (such as the secondary structure; panel a) can simply be over-laid. These two components already perfectly convey the information that high effect regions are mainly found in the transmembrane helices and in close proximity of the binding sites. Displaying this simple fact without a heat map would be daunting due to the high dimensionality of the underlying data.

2.2.3 Multiple sequence alignment viewer

Multiple Sequence Alignment (MSA) is a fundamental approach in modern biological research that arranges nucleotide (DNA/RNA) or amino acid (protein) sequences in a way that captures similarities among them. MSA highlights those sites on biological sequences that evolution has “considered” as important, i.e. sites that have slowly or hardly evolved. Thus, MSAs are essential for the prediction of structural aspects such as secondary structure of proteins (27, 79) and identification of functionally important sites such as binding sites (81). With cross-referencing to phylogeny data, MSAs can be used to understand genomics rearrangements and to identify evolutionary rates (77).

MSAs are also widely used to display complex annotations relating to structure and function, and propagate those to un-annotated sequences (82, 83). Because MSAs are important for annotating biological data, many Bioinformatics databases and services are showing Multiple Sequence Alignments on their web pages. Over the years, a number of MSA viewing solutions were developed and provided to the community. Those included desktop (82, 84) and web (82, 85) applications, as well as libraries that use web technologies to render MSAs (82). Recently, with the ubiquitous adoption of JavaScript as the leading programming language for interactive web application, new JavaScript-based MSA viewing tools were developed and made available (85) .

The MSAViewer is a JavaScript-based visualization tool that can easily be loaded on the web without any previous installation of third party software or library. It renders 1) multiple sequence alignment, 2) sequence logo (86) and 3) phylogenetic tree onto three separate panels. The MSA Viewer was designed to require a small amount of memory and thus delivers a quick performance to the user regardless of the overall size of the alignment it presents. Since its release, the MSA Viewer has been incorporated in a variety of programs: in general purpose tools like the Galaxy (23, 87-89) project (as visualization plugin) and in specialized pipelines like the SNP assay development pipeline from PolyMarker (90, 91).

Through interaction with the viewer, users can intuitively browse and scroll the presentation by simple mouse drag gestures, select rows in the alignment by clicking on their corresponding protein IDs, manipulate and change the presentation by applying all coloring schemes implemented by similar viewers. Figure 7 shows the MSA Viewer component and reviews some of its main features.

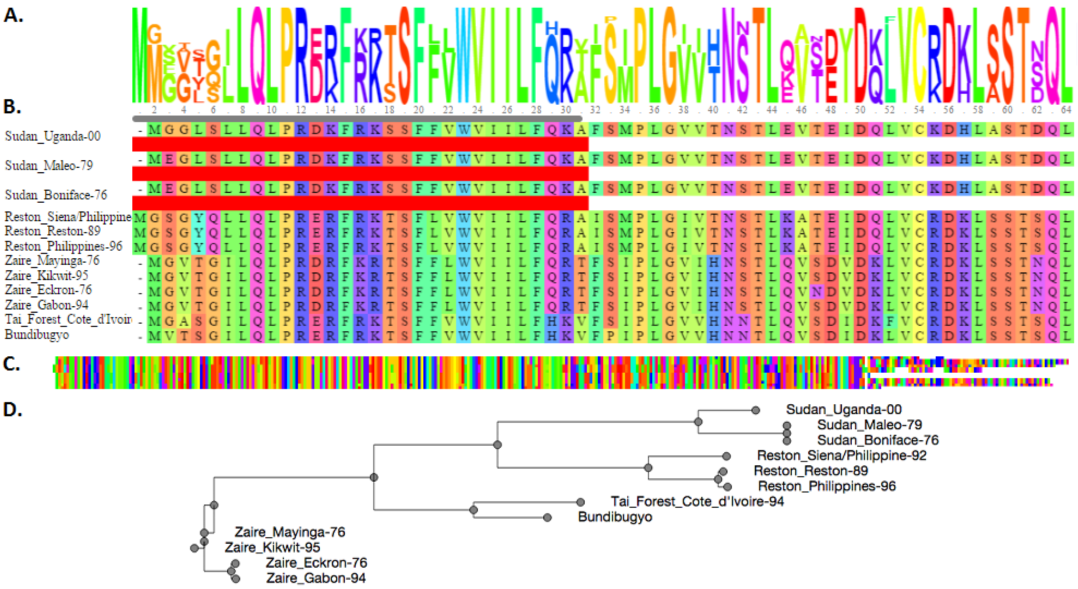


Figure 9 - A simplified view of the MSAViewer for the sequence alignment of the minor nucleoprotein VP 30 within twelve strains of species of virus Filoviridae family. (A) Sequence logo representation with conservation patterns at each position in the MSA. (B) Main MSA panel with residues in the alignment colored according to the default Taylor shading model. Dashes represent gaps. Sequence labeling s below are provided by the user.(92). Red rectangles indicate sequence annotations provided by the user (here: disordered regions as predicted by MetaDisorder (93)). (C) Overview MSA panel with a compact view of the full alignment. (D) Phylogenetic tree representation. Sequence alignment and the phylogenetic tree were calculated using ClustalOmega (94).

2.3 Annotation of proteomes and meta-proteomes

The following section describes the motivation, methodology and results of a project aimed at building a novel service to annotate and analyse proteomes and meta-proteomes.

2.3.1 Background

Microbes carry out metabolic activities that are crucial for environmental processes and the physiology of colonized host species. The gut microbiome is being studied extensively as it has been shown that changes in its composition are correlated with intestinal disease, obesity, immune tolerance and autoimmune diseases(95). Recent advances in molecular techniques, especially high-throughput DNA sequencing and mass spectrometry analysis, revolutionized research on microbial ecosystems and the investigation of key microbial functions in environmental and host-derived ecosystems (96-98). However, the field is now confronted with the fact that the analysis of large datasets remains a major challenge, and the community would greatly benefit from new bioinformatics approaches easing functional interpretation and unraveling the potential in the wealth of published datasets. The most widely used web platforms for 16S rRNA genes and metagenomes analysis are QIIME (99), IMG/M (100) and MG-RAST (101). However, each implementation relies on analysis of single dataset, offering minimal collective meta-analysis capabilities. The Sequence Read Archive (SRA) (102) is the main site of deposition of datasets from sequencing projects (nearing 100,000 archived samples), but offers very limited functionality.

Traditional biological research approaches typically study one gene or a few genes at a time. For example, the Pfam database (25) identifies, marks-up, and characterizes different functional regions within a given protein. The coordinates of these domains are often given in terms of the start and end position within the amino acid sequence. In contrast, high-throughput genomic, proteomic and bioinformatics scanning approaches (such as expression microarray, promoter microarray, proteomics, ChIP-on-CHIPs, etc.) are emerging as alternative technologies that allow investigators to simultaneously measure the changes and regulation of genome-wide genes under certain biological conditions. Those high-throughput technologies usually generate large ‘interesting’ gene lists as their final outputs. Yet the biological interpretation of large, ‘interesting’ gene lists (ranging in size from hundreds to thousands of genes) is still a challenging and daunting task. Over the last few decades, bioinformatics methods, using the biological knowledge accumulated in public databases [e.g. Gene Ontology (10)], make it possible to systematically dissect large gene lists in an attempt to assemble a summary of the most enriched and pertinent biology. State-of-the-art proteome-analysis tools include: DAVID (103), STRING (104) or FFAS (105), but there is room for improvement regarding functional prediction beyond classical annotation and providing user-friendly services.

DAVID (103) is a functional annotation and gene enrichment analysis tool provided by the National Institute of Allergy and Infectious Diseases (NIAID). Over 40 functional categories from dozens of independent public sources (databases) are collected and integrated into the DAVID Knowledgebase. Still those databases fall short in providing functional annotations for hypothetical or sparsely characterized proteins.

The work described in this section discusses the development of a novel resource -- the Rostlab Meta-proteome Annotation Pipeline (RMAP) with the objective to assist functional analysis of proteomics and meta-proteomics data. The pipeline processes multiple datasets of protein sequences, collects annotations form experimental databases and complements those with PredictProtein. Annotations are summarized into a "differential view" i.e. a side-by-side comparison of the functional annotations for each dataset. The pipeline also performs a gene enrichment analysis – a statistical test that ranks overrepresented and underrepresented GO terms.

The pipeline will therefore perform the following tasks:

1. Enable users to upload and label multiple datasets of proteins
2. Automatically annotate the datasets using PredictProtein
3. Summarize annotations and present differential view of datasets
4. Identify enriched GO terms
5. Present enriched GO terms graphically

2.3.2 Datasets

The development of the pipeline was done in collaboration with Dr. Thomas Clavel and his colleagues at the ZIEL Research Center for Nutrition and Food Sciences (TUM, Weihenstephan). Dr. Clavel's research focuses on the study of intestinal bacteria in humans and in mouse models. In a recent (106) study, Clavel et al studied the impact of induced obesity in mice. In the study a group of mice (n=6) subject were fed high-fat (HF) diet for 12 weeks. The gut microbiota of the HF was studies and compared to the gut microbiota of a control group (n=6) fed carbohydrates diet. The study findings show that:

1. The change in diet altered the biochemical composition of the gut microbiota or the activity of bacterial cells
2. The HF diet had the most pronounced impact on pathways of amino acid metabolism
3. Cercal metabolic pathways affected by HF feeding include eicosanoid, steroid hormone, macrolide, bile acid and bilirubin metabolism.

The datasets obtained from Dr. Clavel's group were published together with the gut microbiota study manuscript (106). Proteomics data in the study was acquired through liquid chromatography and tandem mass spectrometry (LC-MS/MS). MS raw data was processed using the Mascot program (107) to identify the set of peptides and proteins. The set of identified proteins were collected into a primary dataset.

The primary dataset contained proteins found in both the HF and control groups. We separated between the proteins in the HF and control samples by averaging the NSAF P-value calculated from the spectral counts of each individual identified protein. NSAF stands for normalized spectral abundance factor and describes the normalized protein abundance in a spectra, i.e. the peptide counts of a protein divided by the protein length and then normalized against all other NSAFs (108). We then cataloged proteins as being part of either the HF or the control group, depending on which group average NSAF value was higher. Since a higher NSAF value corresponds to a higher level of expression, proteins with a higher NSAF value in the HF measurements were sorted into the HF group, the rest belong to the control group. A statistical test was performed on the NSAF values to determine if the differences in expression are statistically significant. Only proteins where this test resulted in a P-value below 0.05 were used in the subsequent analysis. A summary of the datasets sizes is presented in table 1.

Total number of sequences	4,886
HF proteins before filtering	2,403
Control proteins before filtering	2,483
Significant high fat proteins	325
Significant control proteins	517

Table 1 - among the total of 4886 bacterial protein sequences, 2403 proteins were assigned to the HF dataset, and 2483 to the control dataset. After filtering for statistically significant proteins, The HF and CARB datasets consisted of 325 and 517 proteins, respectively. The HF and CARB datasets were studied separately as two distinct sub-datasets. Only the statistically significant" proteins were used in the subsequent analysis.

We also compiled and annotated a reference dataset that includes over 40 sequenced bacterial genomes (Appendix C, list S1) and consists of over 280 thousand proteins. The reference dataset is retrieved from the NCBI protein collection. The NCBI protein collection includes translations from annotated coding regions in GenBank (92), RefSeq (109) and TPA, as well as records from SwissProt (3), PIR (110) PRF, and PDB (2). The reference dataset serves as a background or control against which the genes from the query datasets are enriched. Subsequent discussion refers to the HF and control datasets as query datasets to distinguish those from the reference dataset.

2.3.3 Annotation pipeline

We extended PredictProtein (111) and created a pipeline that annotates arbitrary sized datasets. The workflow currently collects annotations for subcellular localization, GO terms, trans-membrane regions, protein disorder and Pfam (25) functional domains.

After retrieving all sequences from the NCBI protein database the pipeline attempts retrieval of PP annotations stored in the PPcache (section 2.1.6). PPcache allows fast and efficient retrieval of PP predictions without the need to recalculate existing entries. This feature enables the rapid annotation feature of the meta-proteomics annotation pipeline. In case annotations are not found in PPcache the pipeline launches a PredictProtein job for each missing entry. Once all predictions are ready for a given input amino acid sequence the results become available in the PPcache.

When the pipeline has completed annotating all input sequences, results are retrieved from the PPcache, parsed and deposited into a dedicated database. There are nine annotation tables in the database that summarize the annotations for all protein sequences; five tables refer to the query datasets and the other four tables refer to the reference dataset. The annotation tables represent the basis for the functional annotation analysis and comparison between the query datasets and the reference. The query and reference annotation tables for a particular annotation class share the same structure, yet for efficiency reasons the annotations were stored separately. The following annotations are currently being collected, stored in the database and conserved in the analysis:

GO term prediction – protein function annotation is partially based on predictions provided by the metastudent prediction method (section 2.1.7). metastudent maps protein sequences to Gene Ontology terms and provides a set of GO term labels, GO term IDs and reliability measure.

Subcellular localization - following synthesis proteins are sorted to various compartments within the cell to perform their desired function. Determining subcellular localization is therefore important for understanding protein function and is a critical step in proteome annotation. Subcellular-localization annotations are provided by the LocTree3 method (section 2.1.7). LocTree3 takes an amino-acid sequence and uses a set of SVM classifiers combined with homology inference to predict most likely compartment.

Protein disorder - Intrinsically Disordered proteins (IDP) – i.e. proteins that fail to form a stable structure have been shown to play important role in biological activities. It has been claimed by our group that nature uses disordered proteins as a tool to adapt to different environments (112). We tried to put this hypothesis to the test by comparing the number of disordered proteins and disordered regions across the HF and control datasets and comparing both sets to the reference dataset. Prediction of protein disorder are provided by the Meta-Disorder method (section 2.1.7) and (65). The method combines orthogonal approaches for the prediction of protein disorder to a consensus-based prediction algorithm. Following the methodology in (112) we considered a protein to be disordered if the method produced the following classifications:

- *Definition 1*: a short disordered region is a stretch of 30 or more consecutive residues that have been individually predicted to be disordered.
- *Definition 2*: a long disordered region is a stretch of 80 or more consecutive residues that have been individually predicted to be disordered.
- *Definition 3*: an ordered region is a stretch of 50 or more consecutive residues, that have been individually predicted to be ordered.

Subsequently, three different approaches corresponding to the definitions above have been considered:

- Approach 1 (disordered30): identification of short disordered regions; a protein is classified as disordered if and only if it contains at least one short disordered region.
- Approach 2 (disordered80): identification of long disordered regions; a protein is classified as disordered if and only if it contains at least one long disordered region.
- Approach 3 (ordered50): identification of ordered regions; a protein is classified as ordered if it contains at least one ordered region.

Note that most proteins in our sets have been labeled with more than one of the approached described.

Transmembrane proteins - Transmembrane proteins (TPs) can be classified into two types: alpha-helical and beta-barrels. Alpha-helical TPs are present in the inner membranes of bacterial cells and represent the major category of transmembrane proteins. We considered transmembrane protein annotation for two major reasons:

1. Corroborate positive membrane proteins labeling by LocTree3
2. Support significant GO terms predictions by metastudent of membrane specific activity such as transporters and cell signaling.

Transmembrane predictions are provided by TMSEG(41) (Section 2.1.7) a neural network based prediction method.

Statistical Analysis

The final step of the pipeline generates and displays the results for the query dataset against the gold-standard reference, for each annotation class. All graphical charts are created using the R statistical package[4] and illustrate the following statistics:

- Subcellular localization patterns in bacteria - distribution across the query dataset comparing against reference
- Reliability of the subcellular localization predictions - distribution across the query dataset comparing against reference
- Subcellular localization patterns for transmembrane proteins - distribution across the query dataset comparing against reference
- Short disordered protein regions - distribution across the query dataset comparing against reference
- Long disordered protein regions - distribution across the query dataset comparing against reference
- Ordered protein regions - distribution across the query dataset comparing against reference

[4] http://www.r-project.org/

- Overlap between the short and long disordered and ordered protein regions, respectively - parallel distribution across all datasets
- Transmembrane proteins - parallel distribution across all datasets

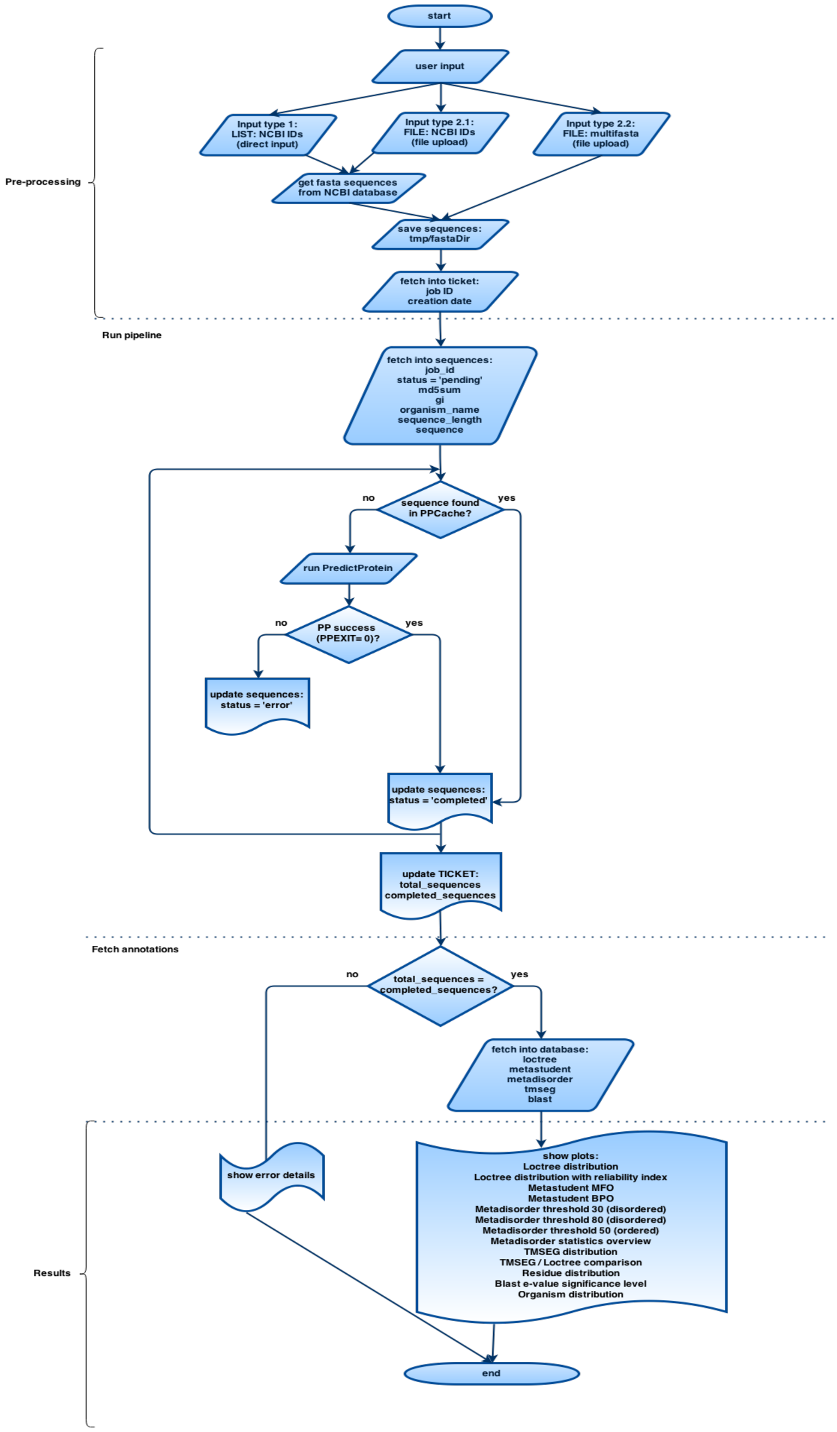

start
user input
Input type 1: LIST: NCBI IDs (direct input)
Input type 2.1: FILE: NCBI IDs (file upload)
Input type 2.2: FILE: multifasta (file upload)
Pre-processing
get fasta sequences from NCBI database
save sequences: tmp/fastaDir
fetch into ticket: job ID creation date
Run pipeline
fetch into sequences: job_id status = 'pending' md5sum gi organism_name sequence_length sequence
no
sequence found in PPCache?
yes
run PredictProtein
no
PP success (PPEXIT= 0)?
yes
update sequences: status = 'error'
update sequences: status = 'completed'
update TICKET: total_sequences completed_sequences
Fetch annotations
no
total_sequences = completed_sequences?
yes
fetch into database: loctree metastudent metadisorder tmseg blast
show error details
show plots: Loctree distribution Loctree distribution with reliability index Metastudent MFO Metastudent BPO Metadisorder threshold 30 (disordered) Metadisorder threshold 80 (disordered) Metadisorder threshold 50 (ordered) Metadisorder statistics overview TMSEG distribution TMSEG / Loctree comparison Residue distribution Blast e-value significance level Organism distribution
Results
end

Figure 10 – schematic representation of the annotation pipeline. Roughly speaking the annotation pipeline is composed of the following steps: pre-processing – collection of user input, gathering of sequences from online resources and job initialization, processing – collection of annotations from the PP cache or via computation, extracting data and deposition into the annotation pipeline database, generating statistics and results visualzations.

2.3.4 Gene enrichment analysis

The pipeline performs a gene annotation enrichment analysis -- a high-throughput strategy that increases the likelihood to identify meaningful biological processes in a list of genes or proteins(113). The most traditional strategy for enrichment analysis is to take a preselected 'interesting' set of genes, and then iteratively test the enrichment of each annotation term one-by-one in a linear mode. Enrichment test is simply done by comparing the frequency of each term. Each term's frequency is then evaluated against the overall distribution of term frequencies. Thereafter, the individual, enriched annotation terms passing the enrichment P-value threshold are reported by the enrichment probability (enrichment P-value). The enrichment P-value calculation, i.e. number of genes in the list that hit a given biology class as compared to pure random chance, can be performed with the aid of some common and well-known statistical methods.

For the gene enrichment analysis in this project we used predictions of Gene Ontology (GO) terms (predicted by Metastudent (53) described in section 2.1.4). The Gene Ontology has three parts: Molecular Function Ontology (MFO), Biological Process Ontology (BPO) and Cellular Component Ontology (CCO). Each of these parts is organized in the GO as a directed acyclic graph and captures different aspects of protein function. Functional keywords ("GO terms") are nodes and their relationships are labeled edges. The ontology is hierarchical: following the edges from a node, each new term corresponds to a more general concept of the original function. All paths converge at the root node, which can simply be interpreted as, e.g., has a molecular function. In this project we used the GO version 1.2, downloaded on 2015-02-24.

The first step in the enrichment analysis is simply counting the number of times each term in the ontology occurs in the query dataset and the reference, respectively. A list of the terms where the difference in frequency in the dataset and the reference is striking can then be generated.

In the next step we test for statistical significance for the differences between the terms observed in the query set as matched against the control set. There are over 68 services and software packages available (113) that perform various types of GO gene enrichment analysis. For this analysis we chose the Biological Network Gene Ontology (BiNGO) tool. BiNGO (114) is a java-based tool that is implemented as a Cytoscape (115) plugin. It implements the hypergeometric and binomial statistical tests to estimate over/under represented GO terms. BiNGO also uses an internal filtering approach to reduce the list of terms by removing redundant GO terms. We found BiNGO to be a versatile tool that could be adapted to our needs mostly because it can be easily used with Cytoscape for presentations. Cytoscape(115) is a well-known platform that is used to integrate, analyze and visualize molecular interaction networks and biological pathways data. We also adapted BiNGO's source code to be able to perform gene enrichment outside of the Cytoscape environment from the command line.

We used BiNGO to asses gene enrichment for both the HF and control sets. The core microbiome set i.e. the reference set was used as a background dataset. We provide the option to use either the Molecular Function Ontology (MFO) or Biological Process Ontology (BPO) parts of the Gene Ontology. We recommend, however, using the MFO since metastudent predictions of MFO terms are more reliable than those of BPO terms.

As mentioned above BiNGO can apply either the hypergeometric or binomial significance tests. The difference between the hypergeometric and binomial tests is simply the distribution the frequencies are assumed to follow. The hypergeometric distribution is analogous to drawing balls from a bin without replacement; the binomial distribution describes the same scenario with replacement.

BiNGO sets the null hypothesis for the hypergeometric test to assume that the probability of a sequence being annotated with the label (term) under consideration is equal in both the reference and the query dataset. It also assumes that the frequency of the annotations follows a certain distribution. BiNGO then calculates the probability of obtaining the given difference between the two frequencies under the above assumptions and the probability then is provided as a P-value measure. The P-value denotes the probability of obtaining the difference in frequencies by random chance. By default, we consider a P-value below 0.05 to be statistically significant.

Because BiNGO tests the significance of all GO terms present in the query set, the number of statistical tests performed in a single analysis may amount to several hundreds. When testing multiple hypotheses, the obtained P-values have to be corrected in order to control the false positive rate (116). To control the expected proportion of false positives among the positively identified tests, known as the False Discover Rate (FDR), BiNGO uses the Benjamini-Hochberg correction procedure (117), which provides strong control over the FDR in that it minimizes the number of false negatives at the cost of a few more false positives.

As a second possibility of combating the multiple-testing problem, BiNGO also implements the Bonferroni correction. The Bonferroni correction works by normalizing the significance threshold of each individual hypothesis by the number of total tests performed. If, for instance, m tests are performed with a significance level of α, the threshold of each individual test would be α/m.

Finally, since BiNGO propagates terms upward in the GO hierarchy, very general terms can appear in the output. These terms, such as 'biosynthetic process' or 'gene expression', are not particularly informative. We thus provide the option to filter out terms that are at a less than a certain distance from the root of the GO ontology.

2.3.5 Results and discussion

Subcellular localization - both the high-fat (HF) and control datasets contain a significantly higher number of proteins that localize to the cytoplasm than the reference set. Furthermore, there was a little change in the percentage of proteins that localize to the cytoplasm when looking at both sets. This may be explained as an artifact of the proteomic mass-spectrometry method, which is overly sensitive to proteins that are involved in translation. Another observation showed that the control dataset contained nearly three times the number of proteins that localize to the inner membrane which may suggest that cell transport activity was somewhat altered as a result from the change in environment.

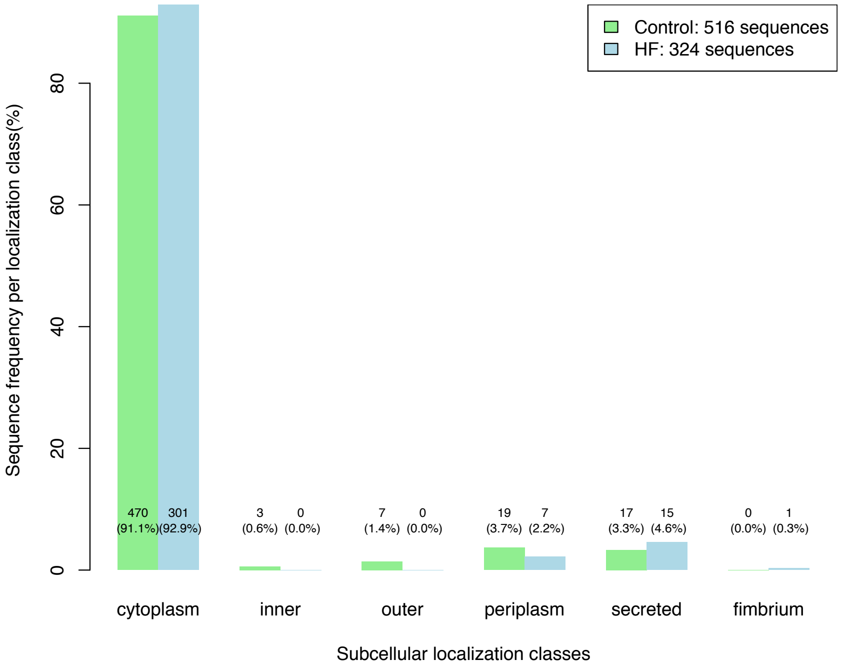


Figure 11 – compariosn of subcellular localizaiton annotation of both query sets (Hf vs. control). Green and blue bars represent control and HF sets respectively. The x-axis lists the six predicted subcellular localziation classes. The ratio of proteins within their dataset (HF or control) per sub cellualr localization class is given on the y-axis. Each bar is labeled with the number of proteins found to localize to each class as well as the ratio of that count within the dataset (expressed as percentage in parenthesis).

Finally as a side effect of this project we encountered an interesting performance measurement relevant to the subcellular localization prediction tool (Locttree3). It seems that while LocTree3 provides medium to high reliable results for the all non-cytoplasm classes, its low reliability (a reliability index between 0-40) shoots up to over 30% for cytoplasmic predictions (figure 12).

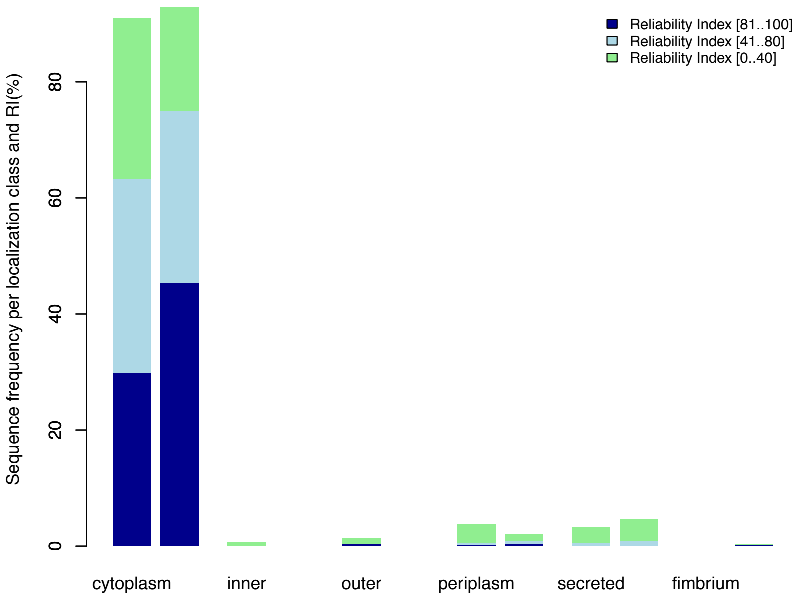
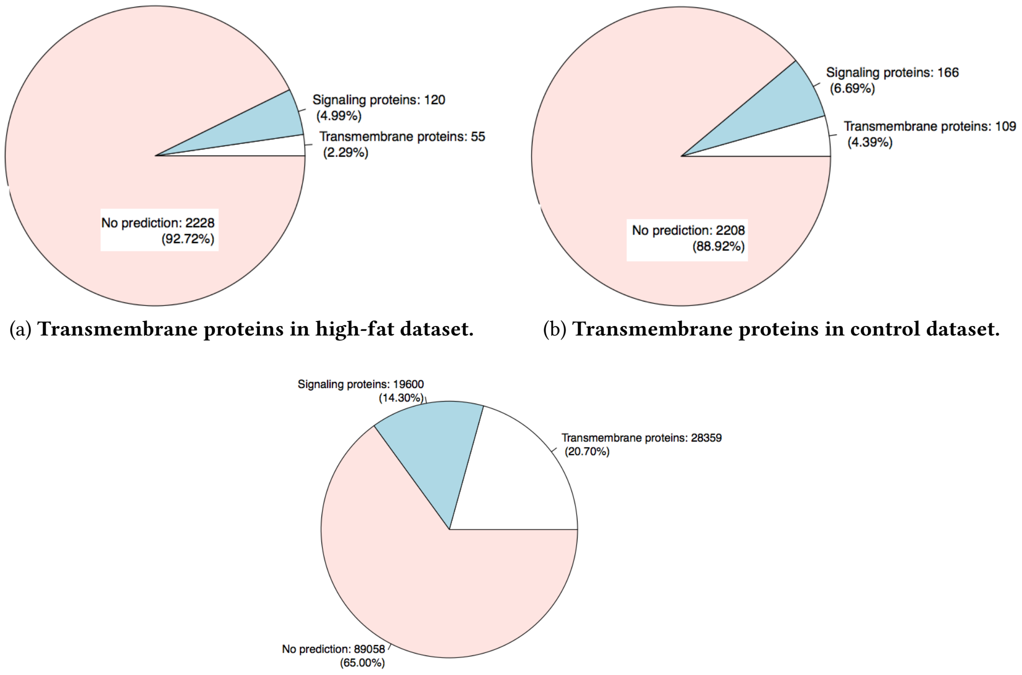


Figure 12 – sub cellular localization prediction summarized for the HF and control dataset. Each bar in the chart shows the number of predicted proteins per localization class and is segemnted by the predictor reliability index. For each pair of bars shown in each localization class the bars on the left expresses the ratio of proteins predicted for that localization for the control dataset. Similiarly bars on the right hand side in each pair express the ratio of proteins predicted to localize in the class labeled on the x-axis for the HF set. The count of most reliable predictions is expressed by blue segment and most unreliable are expressed in green.

Transmembrane regions - another observation regarding the performance of the tools used in this project shows a low degree of overlap between the subcellular localization prediction method (LocTree3) and the transmembrane region predictor TMSEG. On average the two methods provide consistent results (i.e. LocTree3 assign inner/outer membrane prediction to a positive prediction by TMSEG) for approximately 45.5% of the proteins. This observation led us to start a new project of comparing predictions from a set of leading transmembrane regions predictor and intersecting those with LocTree3. Note that since the expression levels found for proteins in each dataset (HF, control) are inconsequential for the performance analysis of the methods we used the entire set of HF and control proteins (2403 and 2483 respectively).



(a) **Transmembrane proteins in high-fat dataset.**

(b) **Transmembrane proteins in control dataset.**

(c) **Transmembrane proteins in reference dataset.**

Figure 13 – summary of TMSEG predictions for the HF set (a), control (b) and refernece (c) datasets as a percentage out of the entire set. No predictions portion of the pie chart is colored pink, signaling proteins colored blue and transmembrane prediction are colored white.

Protein disorder - as the gut bacteria study shows (section 2.3.3) the HF diet alters primarily the bacterial ecosystem at the functional level and less so in the composition of the community, we expected to find a greater number of disordered protein to be expressed in the HF diet than in the control dataset and also at higher levels compared to the reference set. The results from this analysis did not confirm our expectation, namely that the HF set will contain a higher portion of disordered proteins as compared to the reference. The HF dataset did contain a higher portion of disordered proteins than the control (~13% vs ~6%). However, both sets contained a lower portion of disordered proteins compared to the reference set (~20%). One outcome of this analysis was to fine-tune the way we compute the portion of disordered proteins by including a protein expression value (using the NSAF value described in section 2.3.2) as an additional factor in our calculations. This readjustment will take place in a follow up project.

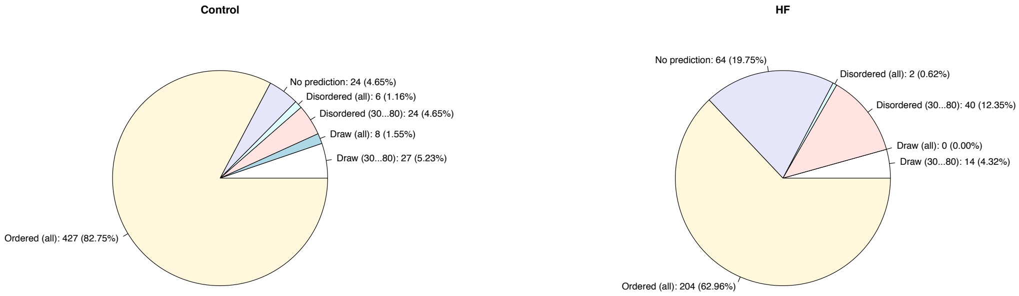


Figure 14 – breakdown of disorder prediction for the control (left pie chart) and HF (right pie chart). Note that disorder classification follows the procedure described in section 2.3.3 (protein disorder). This classification resulted in four disorder classifications: 1. Disordered 30..80 – predictions containing continuous disordered regions between 30 to 08 residues long. 2. Draw 30..80 – predictions containing both ordered and disordered regions between 30 to 80 residues long. 3. Draw all – predictions that fit all classifications. 4. Disordered all – predictions fit all disordered classifications. Note that no assignment (disordered/ordered) is made for proteins labeled as "draw". No predictions proteins in this class contain no ordered regions, no short disordered regions and no long disordered regions.

Gene enrichment analysis

The gut microbiome study demonstrates that "changes in bacterial metaproteome after HF feeding are most pronounced for pathways of amino acid metabolism" (106). In other words, given the deficiency in carbohydrates as an available source for energy production, it has been observed that the bacteria had adapted to the change in environment by altering its metabolic pathway and had now turned to break down amino acids as its primary source for energy production. This finding is supported by the following protein functional analysis protocol (abstract description):

1. Spectra captured through the MS method are aligned against a database of known sequences to identify known proteins in the set.
2. Spectra count is normalized to account for long proteins thus reducing the effect of bias in the set that resulted from the MS method.
3. Identified proteins are functionally classified by aligning them against the COG database (118). Each protein is assigned a COG category.
4. Using the assigned COG categories and the statistical significance of the spectra count for each set, functional differences between the datasets are outlined and reported through gene enrichment analysis.

Note that this procedure relies on the Clusters of Orthologous Groups (COG) database. COG (118) is a popular tool for functional annotation. It relies on complete microbial genomes, which allows reliable assignment of orthologs and paralogs for most genes; Each COG consists of individual orthologous proteins or orthologous sets of paralogs from at least three lineages. Orthologs typically have the same function, allowing transfer of functional information from one member to an entire COG. This relation automatically yields a number of functional predictions for poorly characterized genomes.

In our analysis we replace this workflow with the pipeline described in 2.5.4. Overall our GO term annotation turns 7191 labels in which the HF set was labeled with 1914 terms and the control set was labeled with 5227 terms. It is important to note that the GO Term prediction tool used, Metastudent, labels each protein with multiple predictions, some of them may have direct parent child relationships along the GO graph. As stated earlier we reduce the number of predicted labels considered in the enrichment analysis by eliminating all terms that are catalogued three levels away from the GO graph root. This cutoff value (3) was determined experimentally to remove most of the uninformative terms without affecting terms of interest. BiNGO predicts 113 GO terms to be significantly overrepresented in the HF dataset and 343 in the control set. After filtering out uninformative terms, 103 remain in HF and 301 in control. Finally it is important to note that the reference set has contained 602146 terms and on average there were 4.4 terms predicted per protein.

Overrepresented terms in the HF set are shown in table 3. The terms are sorted by ascending P-value after correction. The most striking overrepresented terms are related to rRNA binding and glutamate dehydrogenase activity. The biological inference of these functions are discussed in more detail later in this section.

GO-ID	Description	P-Value	Corrected P-value	Frequency in Dataset	Frequency in Reference
GO:0019843	rRNA binding	2.99E-22	1.03E-19	33	663
GO:0070181	small ribosomal subunit rRNA binding	3.77E-19	9.74E-17	15	83
GO:0003723	RNA binding	1.53E-18	3.17E-16	76	4848
GO:0004347	glucose-6-phosphate isomerase activity	2.04E-18	3.51E-16	13	55
GO:0004616	phosphogluconate dehydrogenase (decarboxylating) activity	2.78E-16	4.11E-14	13	78
GO:0004354	glutamate dehydrogenase (NADP+) activity	2.19E-13	2.71E-11	9	36
GO:0004353	glutamate dehydrogenase [NAD(P)+] activity	2.88E-13	2.71E-11	9	37
GO:0004352	glutamate dehydrogenase (NAD+) activity	2.88E-13	2.71E-11	9	37
GO:0070728	leucine binding	2.88E-13	2.71E-11	9	37
GO:0016861	intramolecular oxidoreductase activity, interconverting aldoses and ketoses	9.64E-11	8.30E-09	19	536

Table 2 – top 10 overrepresented GO terms in the HF dataset. Calculated with BiNGO using a binomial test with Benjamini correction. Terms with distance at most 3 from the root of the gene ontology were filtered out. Sorted by P-value after correction. The sizes of the dataset and reference are 277 and 52817, respectively. These counts do not include sequences without Metastudent predictions.

GO-ID		P-Value	Corrected P-value	Frequency in Dataset	Frequency in Reference
GO:0003747	translation release factor activity	2.69E-69	4.07E-66	61	267
GO:0006415	translational termination	8.87E-68	1.68E-65	62	297
GO:0043624	cellular protein complex disassembly	8.87E-68	1.68E-65	62	297
GO:0032984	macromolecular complex disassembly	8.87E-68	1.68E-65	62	297
GO:0022411	cellular component disassembly	8.87E-68	1.68E-65	62	297
GO:0071822	protein complex subunit organization	8.87E-68	1.68E-65	62	297
GO:0043241	protein complex disassembly	8.87E-68	1.68E-65	62	297
GO:0008079	translation termination factor activity	8.87E-68	1.68E-65	62	297
GO:0004020	adenylylsulfate kinase activity	4.93E-64	8.28E-62	55	227
GO:0043024	ribosomal small subunit binding	2.63E-57	3.98E-55	55	294

Table 3 – top 10 overrepresented GO terms in the control dataset. Calculated with BiNGO using a binomial test with Benjamini correction. Terms with distance at most 3 from the root of the gene ontology were filtered out. Sorted by P-value after correction. The sizes of the dataset and reference are 436 and 52904, respectively. These counts do not include sequences without Metastudent predictions.

Figures 15-18 show a schematic view of all predicted terms for the HF set and their assigned significance by the gene enrichment analysis. Note that BiNGO connects all terms in the input set by traversing through the GO graph and adding missing terms as nodes in the graphs. The resulting graph then always contains the root element (in the MFO case the root term is molecular function). Terms are laid out in a graph in which nodes represent terms and edges represent parent child relationships between those terms. Each node corresponds exactly to one predicted term and is labeled that term. Nodes vary in size to reflect the level of the GO term they correspond to at the GO hierarchy. Finally, enrichment is represented via node coloring and color intensity. Nodes' colors range varies from pale red to dark red and correspond to the P-value assigned by BiNGO where pale red stand for insignificant and dark red stands for highly significant.

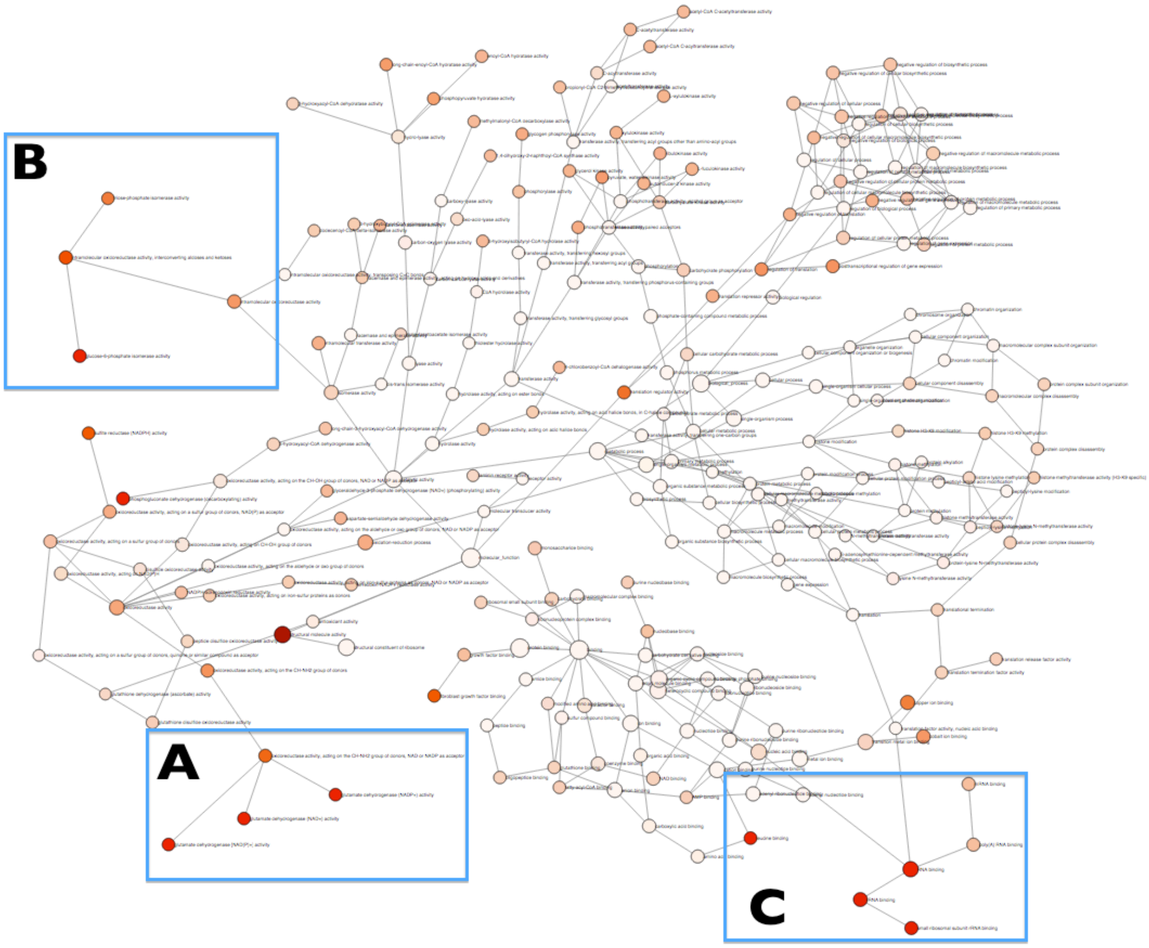

B
ribose-phosphate isomerase activity
intramolecular oxidoreductase activity, interconverting aldoses and ketoses
intramolecular oxidoreductase activity
glucose-6-phosphate isomerase activity
A
oxidoreductase activity, acting on the CH-NH2 group of donors, NAD or NADP as acceptor
glutamate dehydrogenase (NADP+) activity
glutamate dehydrogenase (NAD+) activity
glutamate dehydrogenase [NAD(P)+] activity
C
tRNA binding
poly(A) RNA binding
RNA binding
rRNA binding
small ribosomal subunit rRNA binding
leucine binding

Figure 15 – zoomed out view of all predicted GO terms for the HF set, their P-Value rankings, interrelationships and level position in the GO hierarchy (as reflected by their relative size where smallest nodes are at the leaf level and bigger nodes are at the root level). Blue frames mark areas of interest in the graph and are blown out in subsequent figures. Region in blue frame labeled A is blown out and explain in figure 16; Region in blue frame labeled B is blown out and explained in figure 17; Region in blue frame labeled C is blown out and explained in figure 18.

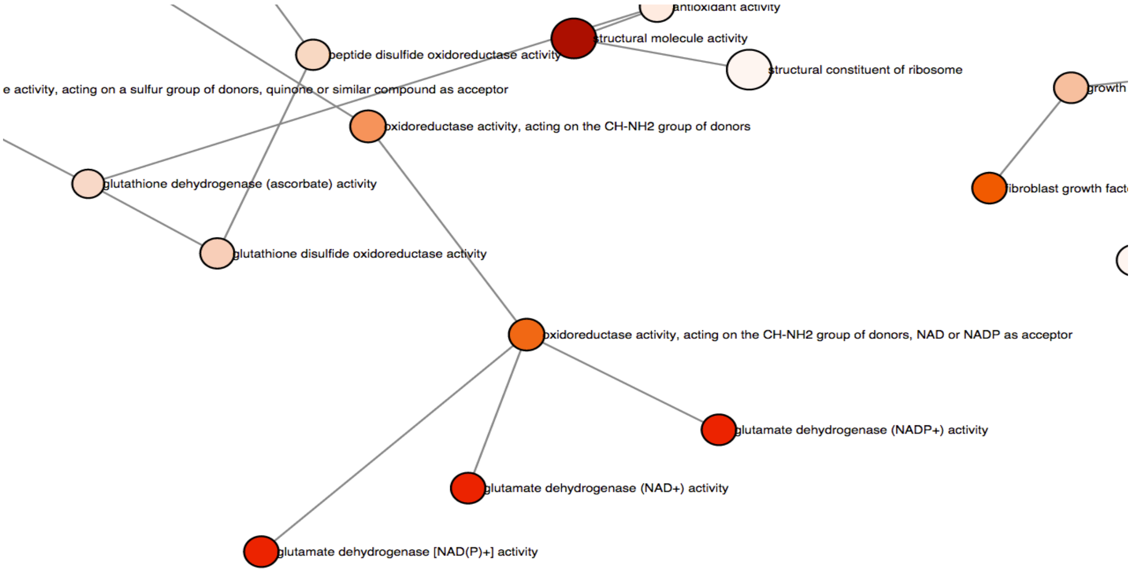

antioxidant activity
tructural molecule activity
eptide disulfide oxidoreductase activity
tructural constituent of ribosome
growth
e activity, acting on a sulfur group of donors, quinone or similar compound as acceptor
xidoreductase activity, acting on the CH-NH2 group of donors
glutathione dehydrogenase (ascorbate) activity
ibroblast growth fact
glutathione disulfide oxidoreductase activity
xidoreductase activity, acting on the CH-NH2 group of donors, NAD or NADP as acceptor
glutamate dehydrogenase (NADP+) activity
glutamate dehydrogenase (NAD+) activity
glutamate dehydrogenase [NAD(P)+] activity

Figure 16 – a blow-out version of the region marked A in figure 15. The figure highlights the over represented group of terms glutamate dehydorgenase activity (parent term is oxidoreductase activity, actng on CH-NH2 group of donors, NAD or NADP as acceptor). The glutemate dehydorgensae activity terms are listed at ranks 3-6 in the list of significantly overrepresented terms among the HF dataset.

We use subcellular localization prediction to support or discard GO term predictions. As an example, the proteins labeled with the glutamate dehydrogenase term were also predicted by LocTree3 to localize to the cytosol as expected.

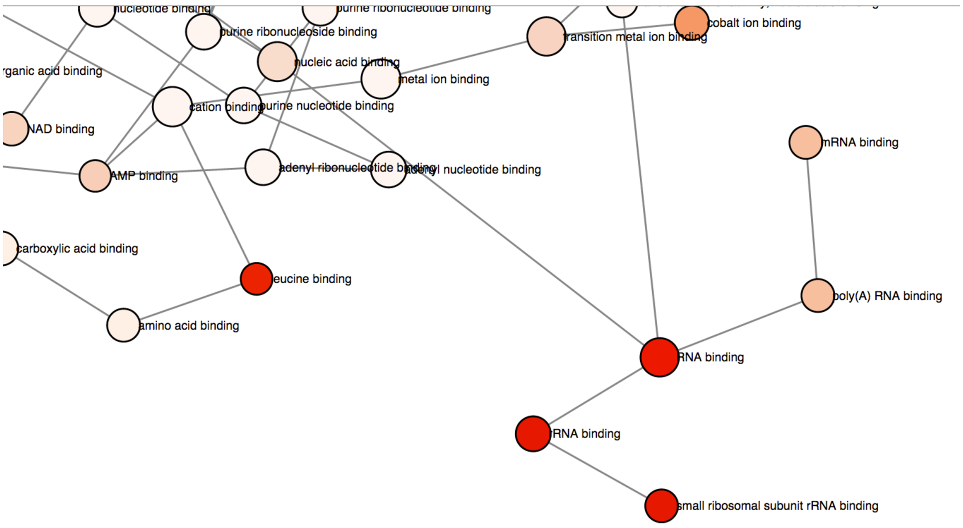


Figure 17 - a blow-out version of the region marked C in figure 15. The figure highlights the three related terms rRNA binding, small ribosomal subunit rRNA binding, rRNA binding respectively positioned at first, second and third places in the list of significantly overrepresented terms among the HF dataset. Note that Leucine binding also appears in the figure yet the term is filtered out by our algorithm as uninformative.

The enrichment analysis also highlighted ribosomal RNA binding as en enriched activity in the HF diet (Figure 14). Translation is the net effect of proteins being synthesized by ribosome in the cytosol. The mouse microbiome study findings show that the HF diet-induced increase in spectral abundance factors related to COG category J (translation). Authors of the study postulate that this may reflect adaptation of microbial cells to meet their needs for survival in a milieu with low energy originating from carbohydrates.

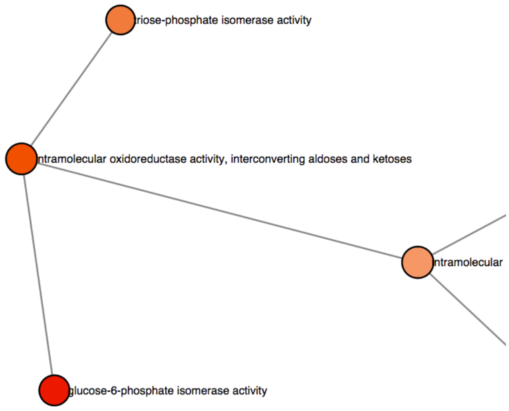

riose-phosphate isomerase activity
ntramolecular oxidoreductase activity, interconverting aldoses and ketoses
ntramolecular
lucose-6-phosphate isomerase activity

Figure 18 - a blow-out version of the region marked B in figure 15. The figure highlights the GO term glucose-6-phospate isomerase activity ranked fourth in the list of significantly overrepresented terms among the HF dataset.

We also note that the gene enrichment analysis highlighted the glucose-6-phophate isomerase as an enriched activity in the HF diet. This seems to be inconsistent with the underlying data as the glucose-6-phophate isomerase enzyme fulfills a central role in the glycolysis pathway by converting glucose into fructose and the activity is not expected to be enriched in glucose deficient environment. Mismatches such as this example can point to the weakness of the underlying approach (relying on a prediction method as a primary source) and highlights the need for a careful cross-referencing.

2.3.6 Web server

The RMAP web interface provides an input box onto which users can upload two separate lists of GI ids representing their datasets. Upon submission the server maps the GIs to protein sequence using the NCBI server (http://www.ncbi.nlm.nih.gov/). A number of reports are then generated from the annotations. These are described more closely in section 2.3.5. The statistics are visualized using interactive JavaScript plots. The results of the gene enrichment analysis are illustrated in two different ways. A table view shows the significantly enriched terms in a simple list with P-values and links to more detailed descriptions on the Gene Ontology website. The other view displays the terms in a graph. Individual entries in the table or graph can be clicked to show a list of the proteins that are annotated with the corresponding GO term, along with descriptions and links to further resources.

3 Conclusion

The scope of the work presented here was the creation of systems that provide a battery of prediction methods that help bridge the sequence annotation gap. It has been observed in this work that in order to be able to provide systems that will meet the demands of modern biology, those systems need to become resilient, scalable and flexible so they will be able to deliver data in a consistent, unified and accessible manner.

In the first part of this work we focused on applying the time tested best practices of software engineering. We meticulously re-organized the code base of dozens of software tools and arranged them into software packages. This in turn allowed us to create a stable, scalable and predictable system that withstands the challenge of constant change. We made the software available in open source form to benefit from a community of developers that will be able to extend our tools and improve on them. We also created a cloud-ready version of he software so it could be easily scaled and be used in high throughput experiments. The lessons learned from this work and the recommendation gathered in this thesis can be applied to any bioinformatics system.

Next, we leveraged on the improvements introduced in the first step and constructed a new and expanded version of the PredictProtein protein structure and function online service. User experience has been enhanced by the addition of the PredictProtein cache which delivers faster response time to ~45% of the jobs being submitted to the server. User experience has also been enhanced by the introduction of visualPP, the web application that unifies the presentation of results into a set of interactive images. visualPP helps researchers better understand the predictions we provide. Creating visualPP also benefitted the community through the contribution of several visualization tools such as the HeatMapViewer and the MSAViewer that provide a readily available visualization solutions to present data types used in everyday biology.

In the final part of this work we extended the PredictProtein software suite and used the PPCache to build a prototype system for the annotation of proteomic and meta-proteomic data. The novelty of the system is its capability to summarize protein function annotations and show a differential view across datasets of interest. Unlike other systems that use experimental data to preform gene enrichment, the proposed annotation pipeline uses predictions and thus is well suited for studies that focus on multiple uncharacterized organisms. Through the analysis of experimental datasets derived from the mouse gut meta-proteome, we have shown that such a system may become useful. Future work will have to expand the array of statistical methods employed by this pipeline.

4 Acknowledgments

It has been a great honor and pleasure to be a member of the Rostlab starting at Columbia University in New York and then moving to the Technical University in Munich. Throughout my time in the group I have had the privilege to work with so many bright, creative and talented people from whom I have learnt and who I am happy to call my colleagues and friends.

First and foremost I owe my gratitude to my supervisor Burkhard Rost who have taken me into his group and have taught me about science, computers, history, art, politics and so much more. Burkhard has afforded me freedom and flexibility in my work and helped me adopt an open-minded approach to problem solving both in my work and in everyday life. I had many enjoyable and insightful conversations with Burkhard who helped me think differently, which has been very valuable to me. Burkhard always encouraged me to pursue my goals and aspirations and this work could not have happened without his support.

I am also very thankful to Yana Bromberg who agreed to co-supervise this work and to host me in her lab at Rutgers. Yana is all that I could ask for in a supervisor – she is amazingly quick to grasp the problems at hand and she always have great suggestions on how to look at data and deal with the inevitable analysis challenges that come up. Yana always welcomed me with a cheerful smile and showed a great deal of patience and willingness to guide the progress of this work.

I am thankful for still keeping in touch with my old Columbia colleagues and friends Yanay Ofran and Avner Schlessigner. Both Yanay and Avner are brilliant scientists whose insight and advice helped me in my work.

Thanks to all Rostlab members past and present. Especially to Laszlo Kajan who I had the pleasure to work with closely and who have taught me a great deal about computers, programming and bringing an analytical and critical thinking approach to my work. Sebastian Wilzbach and Tatyana Goldberg are great colleagues and friends whom I enjoyed working with throughout the BioJS and GSoC experience. Thanks guys you're awesome! I wish to also thank Edda Kloppman with whom I had the pleasure to work with in several instances where I learned from her professionalism and dedication. Thanks to Juan-Miguel Cejuela and Pandu Raharja-Liu who I had the pleasure to work with on the PubSeq project. Thanks to Tobias Hamp for his helpful advice on gene enrichment analysis. To Maximillian Hecht who worked with me on the HeatMap viewer. Thanks to Lothar Richter for helping with administrative issues. Special thanks to my bachelor students, Diana Iaacob and Jonas Raedle, who worked tirelessly and with great enthusiasm on the meta-proteome analysis pipeline project. To Tim Karl for all of his help with IT matters and for collaborating on many projects. I am indebted to Marlena Drabik who greeted me on my first visit to Munich and since then has been an indispensible source of help that help me navigate Munich as well as the Technical University. My heartfelt thanks to Manuela Fischer from the dean's office who without her patience and help I would have never managed to get to complete this work.

On a personal note I wish to thank my family and friends for their endless love and support. To my parents Yehudit and Yaakov Yachdav and to my sisters Einav Yachdav-Camerini and Nofar Yachdav-Malka for backing me up in every decision I made and for everything they taught me. I am thankful for my wonderful boys Ilay and Ari who have been a source of inspiration and strength. Last but not least to my loving wife, Galit who has been supporting and encouraging me to pursue a Ph.D. at this stage of my life. I couldn't have done any of this without her.

References

1. Redfern OC, Dessailly B, Orengo CA. Exploring the structure and function paradigm. Curr Opin Struct Biol. 2008;18(3):394-402. doi: 10.1016/j.sbi.2008.05.007. PubMed PMID: 18554899; PMCID: 2561214.

2. Rose PW, Bi C, Bluhm WF, Christie CH, Dimitropoulos D, Dutta S, Green RK, Goodsell DS, Prlic A, Quesada M, Quinn GB, Ramos AG, Westbrook JD, Young J, Zardecki C, Berman HM, Bourne PE. The RCSB Protein Data Bank: new resources for research and education. Nucleic acids research. 2013;41(Database issue):D475-82. Epub 2012/11/30. doi: 10.1093/nar/gks1200. PubMed PMID: 23193259; PMCID: 3531086.

3. Magrane M, Consortium U. UniProt Knowledgebase: a hub of integrated protein data. Database (Oxford). 2011;2011:bar009. Epub 2011/03/31. doi: 10.1093/database/bar009. PubMed PMID: 21447597; PMCID: 3070428.

4. Mardis ER. A decade's perspective on DNA sequencing technology. Nature. 2011;470(7333):198-203. doi: 10.1038/nature09796. PubMed PMID: 21307932.

5. Stein LD. The case for cloud computing in genome informatics. Genome Biol. 2010;11(5):207. doi: 10.1186/gb-2010-11-5-207. PubMed PMID: 20441614; PMCID: 2898083.

6. Metzker ML. Sequencing technologies - the next generation. Nature reviews Genetics. 2010;11(1):31-46. doi: 10.1038/nrg2626. PubMed PMID: 19997069.

7. The human genome at ten. Nature. 2010;464(7289):649-50. doi: 10.1038/464649a. PubMed PMID: 20360688.

8. KA. W. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP) [cited 2015 4/1/2015]. Available from: http://www.genome.gov/sequencingcosts.

9. Reddy TB, Thomas AD, Stamatis D, Bertsch J, Isbandi M, Jansson J, Mallajosyula J, Pagani I, Lobos EA, Kyrpides NC. The Genomes OnLine Database (GOLD) v.5: a metadata management system based on a four level (meta)genome project classification. Nucleic Acids Res. 2015;43(Database issue):D1099-106. doi: 10.1093/nar/gku950. PubMed PMID: 25348402; PMCID: 4384021.

10. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S,

Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet. 2000;25(1):25-9. Epub 2000/05/10. doi: 10.1038/75556. PubMed PMID: 10802651; PMCID: 3037419.

11. O'Donovan C, Apweiler R, Bairoch A. The human proteomics initiative (HPI). Trends in biotechnology. 2001;19(5):178-81. PubMed PMID: 11301130.

12. CASP11 in numbers 2014 [cited 2015 30/03]. Available from: http://www.predictioncenter.org/casp11/numbers.cgi.

13. Moult J, Fidelis K, Rost B, Hubbard T, Tramontano A. Critical assessment of methods of protein structure prediction (CASP)--round 6. Proteins. 2005;61 Suppl 7:3-7. doi: 10.1002/prot.20716. PubMed PMID: 16187341.

14. Szklarczyk D, Franceschini A, Wyder S, Forslund K, Heller D, Huerta-Cepas J, Simonovic M, Roth A, Santos A, Tsafou KP, Kuhn M, Bork P, Jensen LJ, von Mering C. STRING v10: protein-protein interaction networks, integrated over the tree of life. Nucleic Acids Res. 2015;43(Database issue):D447-52. doi: 10.1093/nar/gku1003. PubMed PMID: 25352553; PMCID: 4383874.

15. Bromberg Y, Rost B. SNAP: predict effect of non-synonymous polymorphisms on function. Nucleic acids research. 2007;35(11):3823-35. Epub 2007/05/29. doi: 10.1093/nar/gkm238. PubMed PMID: 17526529; PMCID: 1920242.

16. Bromberg Y, Yachdav G, Rost B. SNAP predicts effect of mutations on protein function. Bioinformatics. 2008;24(20):2397-8. Epub 2008/09/02. doi: 10.1093/bioinformatics/btn435. PubMed PMID: 18757876; PMCID: 2562009.

17. Devereux J, Haeberli P, Smithies O. A comprehensive set of sequence analysis programs for the VAX. Nucleic Acids Res. 1984;12(1 Pt 1):387-95. PubMed PMID: 6546423; PMCID: 321012.

18. Henikoff S. Sequence analysis by electronic mail server. Trends in Biochemical Sciences. 1993;18:267-8.

19. Smith RF, Wiese BA, Wojzynski MK, Davison DB, Worley KC. BCM Search Launcher--an integrated interface to molecular biology data base search and analysis services available on the World Wide Web. Genome research. 1996;6(5):454-62. PubMed PMID: 8743995.

20. Dowell RD, Jokerst RM, Day A, Eddy SR, Stein L. The distributed annotation system. BMC Bioinformatics. 2001;2:7. PubMed PMID: 11667947; PMCID: 58584.

21. Stevens RD, Robinson AJ, Goble CA. myGrid: personalised bioinformatics on the information grid. Bioinformatics. 2003;19 Suppl 1:i302-4. PubMed PMID: 12855473.

22. Finn RD, Stalker JW, Jackson DK, Kulesha E, Clements J, Pettett R. ProServer: a simple, extensible Perl DAS server. Bioinformatics. 2007;23(12):1568-70. doi: 10.1093/bioinformatics/btl650. PubMed PMID: 17237073; PMCID: 2989875.

23. Boekel J, Chilton JM, Cooke IR, Horvatovich PL, Jagtap PD, Kall L, Lehtio J, Lukasse P, Moerland PD, Griffin TJ. Multi-omic data analysis using Galaxy. Nat Biotechnol. 2015;33(2):137-9. doi: 10.1038/nbt.3134. PubMed PMID: 25658277.

24. Biegert A, Mayer C, Remmert M, Soding J, Lupas AN. The MPI Bioinformatics Toolkit for protein sequence analysis. Nucleic Acids Res. 2006;34(Web Server issue):W335-9. doi: 10.1093/nar/gkl217. PubMed PMID: 16845021; PMCID: 1538786.

25. Punta M, Coggill PC, Eberhardt RY, Mistry J, Tate J, Boursnell C, Pang N, Forslund K, Ceric G, Clements J, Heger A, Holm L, Sonnhammer EL, Eddy SR, Bateman A, Finn RD. The Pfam protein families database. Nucleic acids research. 2012;40(Database issue):D290-301. Epub 2011/12/01. doi: 10.1093/nar/gkr1065. PubMed PMID: 22127870; PMCID: 3245129.

26. Jones DT. Improving the accuracy of transmembrane protein topology prediction using evolutionary information. Bioinformatics. 2007;23(5):538-44. Epub 2007/01/24. doi: 10.1093/bioinformatics/btl677. PubMed PMID: 17237066.

27. Buchan DW, Minneci F, Nugent TC, Bryson K, Jones DT. Scalable web services for the PSIPRED Protein Analysis Workbench. Nucleic Acids Res. 2013;41(Web Server issue):W349-57. doi: 10.1093/nar/gkt381. PubMed PMID: 23748958; PMCID: 3692098.

28. Alberts B JA, Lewis J, et al. Molecular Biology of the Cell. 4 ed. New York: Garland Science; 2002.

29. Kendrew JC. Structure and function in myoglobin and other proteins. Federation proceedings. 1959;18(2, Part 1):740-51. PubMed PMID: 13672267.

30. Anfinsen CB. Principles that govern the folding of protein chains. Science. 1973;181(4096):223-30. PubMed PMID: 4124164.

31. Vucetic S, Xie H, Iakoucheva LM, Oldfield CJ, Dunker AK, Obradovic Z, Uversky VN. Functional anthology of intrinsic disorder. 2. Cellular components, domains, technical terms, developmental processes, and coding sequence diversities correlated

with long disordered regions. Journal of proteome research. 2007;6(5):1899-916. doi: 10.1021/pr060393m. PubMed PMID: 17391015; PMCID: 2588346.

32. Dunker AK, Brown CJ, Lawson JD, Iakoucheva LM, Obradovic Z. Intrinsic disorder and protein function. Biochemistry. 2002;41(21):6573-82. PubMed PMID: 12022860.

33. Corpas M, Jimenez R, Carbon SJ, Garcia A, Garcia L, Goldberg T, Gomez J, Kalderimis A, Lewis SE, Mulvany I, Pawlik A, Rowland F, Salazar G, Schreiber F, Sillitoe I, Spooner WH, Thanki AS, Villaveces JM, Yachdav G, Hermjakob H. BioJS: an open source standard for biological visualisation - its status in 2014. F1000Res. 2014;3:55. doi: 10.12688/f1000research.3-55.v1. PubMed PMID: 25075290; PMCID: 4103492.

34. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol. 1993;232:584-99.

35. Rost B. PHD: predicting one-dimensional protein structure by profile-based neural networks. Methods Enzymol. 1996;266:525-39. Epub 1996/01/01. PubMed PMID: 8743704.

36. rostlab. The PredictProtein Team 2013 [cited 2015 04/22/2015]. Available from: https://predictprotein.org/credits.

37. Leprevost Fda V, Barbosa VC, Francisco EL, Perez-Riverol Y, Carvalho PC. On best practices in the development of bioinformatics software. Frontiers in genetics. 2014;5:199. doi: 10.3389/fgene.2014.00199. PubMed PMID: 25071829; PMCID: 4078907.

38. Kajan L, G Y. Packages Munich: Rostlab; 2014 [cited 2015 04/13/2015]. Available from: https://rostlab.org/owiki/index.php/Packages.

39. Ince DC, Hatton L, Graham-Cumming J. The case for open computer programs. Nature. 2012;482(7386):485-8. doi: 10.1038/nature10836. PubMed PMID: 22358837.

40. Ofran Y, Rost B. ISIS: interaction sites identified from sequence. Bioinformatics. 2007;23(2):e13-6. Epub 2007/01/24. doi: 10.1093/bioinformatics/btl303. PubMed PMID: 17237081.

41. Bernhofer M. Transmembrane helix prediction in proteins [Master]. Munich, Germany: Technische Universitaet Muenchen (TUM); 2014.

42. Tusnady GE, Dosztanyi Z, Simon I. PDB_TM: selection and membrane localization of transmembrane proteins in the protein data bank. Nucleic acids

research. 2005;33(Database issue):D275-8. Epub 2004/12/21. doi: 10.1093/nar/gki002. PubMed PMID: 15608195; PMCID: 539956.

43. Lomize MA, Lomize AL, Pogozheva ID, Mosberg HI. OPM: orientations of proteins in membranes database. Bioinformatics. 2006;22(5):623-5. Epub 2006/01/07. doi: 10.1093/bioinformatics/btk023. PubMed PMID: 16397007.

44. Petersen TN, Brunak S, von Heijne G, Nielsen H. SignalP 4.0: discriminating signal peptides from transmembrane regions. Nat Methods. 2011;8(10):785-6. Epub 2011/10/01. doi: 10.1038/nmeth.1701. PubMed PMID: 21959131.

45. Kall L, Krogh A, Sonnhammer EL. An HMM posterior decoder for sequence feature prediction that includes homology information. Bioinformatics. 2005;21 Suppl 1:i251-7. Epub 2005/06/18. doi: 10.1093/bioinformatics/bti1014. PubMed PMID: 15961464.

46. Hecht M. Improve predictions of functional effect of non-synonymous SNPs. Munich, Germany: Technische Universitaet Muenchen (TUM); 2011.

47. Bromberg Y, Overton J, Vaisse C, Leibel RL, Rost B. In silico mutagenesis: a case study of the melanocortin 4 receptor. Faseb J. 2009;23(9):3059-69. Epub 2009/05/07. doi: 10.1096/fj.08-127530. PubMed PMID: 19417090; PMCID: 2735358.

48. Hecht M, Bromberg Y, Rost B. News from the protein mutability landscape. J Mol Biol. 2013;425(21):3937-48. Epub 2013/07/31. doi: 10.1016/j.jmb.2013.07.028. PubMed PMID: 23896297.

49. Yachdav G, Hecht M, Pasmanik-Chor M, al. e. HeatMapViewer: interactive display of 2D data in biology. F1000Research. 2014;3(48). doi: 10.12688/f1000research.3-48.v1.

50. Goldberg T, Hecht M, Hamp T, Karl T, Yachdav G, Ahmed N, Altermann U, Angerer P, Ansorge S, Balasz K, Bernhofer M, Betz A, Cizmadija L, Trinh K, Gerke J, Greil R, Joerdens V, Hastreiter M, Hembach a, Herzog M, Kalemanov M, Kluge M, Meier A, Nasir H, Neumaier U, Prade V, Reeb J, Sorokoumov A, Troshani I, Vorberg S, Waldraff S, Zierer J, Nielsen H, Rost B. LocTree3 prediction of localization. Nucleic Acids Research. 2014.

51. Goldberg T, Hamp T, Rost B. LocTree2 predicts localization for all domains of life. Bioinformatics. 2012;28(18):i458-i65. Epub 2012/09/11. doi: 10.1093/bioinformatics/bts390. PubMed PMID: 22962467; PMCID: 3436817.

52. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic acids research. 1997;25(17):3389-402. Epub 1997/09/01. PubMed PMID: 9254694; PMCID: 146917.

53. Hamp T, Kassner R, Seemayer S, Vicedo E, Schaefer C, Achten D, Auer F, Boehm A, Braun T, Hecht M, Heron M, Honigschmid P, Hopf TA, Kaufmann S, Kiening M, Krompass D, Landerer C, Mahlich Y, Roos M, Rost B. Homology-based inference sets the bar high for protein function prediction. BMC Bioinformatics. 2013;14 Suppl 3:S7. Epub 2013/03/27. doi: 10.1186/1471-2105-14-S3-S7. PubMed PMID: 23514582; PMCID: 3584931.

54. Bairoch A, Boeckmann B, Ferro S, Gasteiger E. Swiss-Prot: juggling between evolution and stability. Brief Bioinform. 2004;5(1):39-55. Epub 2004/05/22. PubMed PMID: 15153305.

55. Radivojac P, Clark WT, Oron TR, Schnoes AM, Wittkop T, Sokolov A, Graim K, Funk C, Verspoor K, Ben-Hur A, Pandey G, Yunes JM, Talwalkar AS, Repo S, Souza ML, Piovesan D, Casadio R, Wang Z, Cheng J, Fang H, Gough J, Koskinen P, Toronen P, Nokso-Koivisto J, Holm L, Cozzetto D, Buchan DW, Bryson K, Jones DT, Limaye B, Inamdar H, Datta A, Manjari SK, Joshi R, Chitale M, Kihara D, Lisewski AM, Erdin S, Venner E, Lichtarge O, Rentzsch R, Yang H, Romero AE, Bhat P, Paccanaro A, Hamp T, Kassner R, Seemayer S, Vicedo E, Schaefer C, Achten D, Auer F, Boehm A, Braun T, Hecht M, Heron M, Honigschmid P, Hopf TA, Kaufmann S, Kiening M, Krompass D, Landerer C, Mahlich Y, Roos M, Bjorne J, Salakoski T, Wong A, Shatkay H, Gatzmann F, Sommer I, Wass MN, Sternberg MJ, Skunca N, Supek F, Bosnjak M, Panov P, Dzeroski S, Smuc T, Kourmpetis YA, van Dijk AD, ter Braak CJ, Zhou Y, Gong Q, Dong X, Tian W, Falda M, Fontana P, Lavezzo E, Di Camillo B, Toppo S, Lan L, Djuric N, Guo Y, Vucetic S, Bairoch A, Linial M, Babbitt PC, Brenner SE, Orengo C, Rost B, Mooney SD, Friedberg I. A large-scale evaluation of computational protein function prediction. Nat Methods. 2013;10(3):221-7. Epub 2013/01/29. doi: 10.1038/nmeth.2340. PubMed PMID: 23353650; PMCID: 3584181.

56. Minneci F, Piovesan D, Cozzetto D, Jones DT. FFPred 2.0: improved homology-independent prediction of gene ontology terms for eukaryotic protein sequences. PLoS One. 2013;8(5):e63754. Epub 2013/05/30. doi: 10.1371/journal.pone.0063754. PubMed PMID: 23717476; PMCID: 3661659.

57. Schlessinger A, Punta M, Rost B. Natively unstructured regions in proteins identified from contact predictions. Bioinformatics. 2007;23(18):2376-84. Epub 2007/08/22. doi: 10.1093/bioinformatics/btm349. PubMed PMID: 17709338.

58. Punta M, Rost B. PROFcon: novel prediction of long-range contacts. Bioinformatics. 2005;21(13):2960-8. Epub 2005/05/14. doi: 10.1093/bioinformatics/bti454. PubMed PMID: 15890748.

59. Schlessinger A, Liu J, Rost B. Natively unstructured loops differ from other loops. PLoS Comput Biol. 2007;3(7):e140. Epub 2007/07/31. doi: 10.1371/journal.pcbi.0030140. PubMed PMID: 17658943; PMCID: 1924875.

60. Liu J, Tan H, Rost B. Loopy proteins appear conserved in evolution. J Mol Biol. 2002;322(1):53-64. Epub 2002/09/07. PubMed PMID: 12215414.

61. Liu J, Rost B. NORSp: predictions of long regions without regular secondary structure. Nucleic Acids Research. 2003;31(13):3833-5.

62. Berman HM. The Protein Data Bank: a historical perspective. Acta Crystallogr A. 2008;64(Pt 1):88-95. Epub 2007/12/25. doi: S0108767307035623 [pii]

10.1107/S0108767307035623. PubMed PMID: 18156675.

63. Schlessinger A, Rost B. Protein flexibility and rigidity predicted from sequence. Proteins. 2005;61(1):115-26. Epub 2005/08/05. doi: 10.1002/prot.20587. PubMed PMID: 16080156.

64. Schlessinger A, Yachdav G, Rost B. PROFbval: predict flexible and rigid residues in proteins. Bioinformatics. 2006;22(7):891-3. Epub 2006/02/04. doi: 10.1093/bioinformatics/btl032. PubMed PMID: 16455751.

65. Schlessinger A, Punta M, Yachdav G, Kajan L, Rost B. Improved disorder prediction by combination of orthogonal approaches. PLoS One. 2009;4(2):e4433. Epub 2009/02/12. doi: 10.1371/journal.pone.0004433. PubMed PMID: 19209228; PMCID: 2635965.

66. Dosztanyi Z, Csizmok V, Tompa P, Simon I. IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. Bioinformatics. 2005;21(16):3433-4. Epub 2005/06/16. doi: 10.1093/bioinformatics/bti541. PubMed PMID: 15955779.

67. Ward JJ, McGuffin LJ, Bryson K, Buxton BF, Jones DT. The DISOPRED server for the prediction of protein disorder. Bioinformatics. 2004;20(13):2138-9. Epub 2004/03/27. doi: 10.1093/bioinformatics/bth195. PubMed PMID: 15044227.

68. Mizianty MJ, Stach W, Chen K, Kedarisetti KD, Disfani FM, Kurgan L. Improved sequence-based prediction of disordered regions with multilayer fusion of multiple

information sources. Bioinformatics. 2010;26(18):i489-96. Epub 2010/09/09. doi: 10.1093/bioinformatics/btq373. PubMed PMID: 20823312; PMCID: 2935446.

69. Sickmeier M, Hamilton JA, LeGall T, Vacic V, Cortese MS, Tantos A, Szabo B, Tompa P, Chen J, Uversky VN, Obradovic Z, Dunker AK. DisProt: the Database of Disordered Proteins. Nucleic Acids Res. 2007;35(Database issue):D786-93. PubMed PMID: 17145717.

70. Hamp T, Rost B. Alternative protein-protein interfaces are frequent exceptions. PLoS Comput Biol. 2012;8(8):e1002623. Epub 2012/08/10. doi: 10.1371/journal.pcbi.1002623. PubMed PMID: 22876170; PMCID: 3410849.

71. Hönigschmid P. Improvement of DNA- and RNA-Protein Binding Prediction: Technische Universität München; 2012.

72. Ashkenazy H, Erez E, Martz E, Pupko T, Ben-Tal N. ConSurf 2010: calculating evolutionary conservation in sequence and structure of proteins and nucleic acids. Nucleic acids research. 2010;38(Web Server issue):W529-33. Epub 2010/05/19. doi: 10.1093/nar/gkq399. PubMed PMID: 20478830; PMCID: 2896094.

73. Celniker G, Nimrod G, Ashkenazy H, Glaser F, Martz E, Mayrose I, Pupko T, Ben-Tal N. ConSurf: Using Evolutionary Data to Raise Testable Hypotheses about Protein Function. Israel Journal of Chemistry. 2013;53(3-4):199-206. doi: 10.1002/ijch.201200096.

74. Rost B, Liu J, Nair R, Wrzeszczynski KO, Ofran Y. Automatic prediction of protein function. Cellular and Molecular Life Sciences. 2003;60(12):2637-50.

75. Przybylski D, Rost B. Alignments grow, secondary structure prediction improves. Proteins. 2002;46(2):197-205. Epub 2002/01/25. PubMed PMID: 11807948.

76. Mayrose I, Graur D, Ben-Tal N, Pupko T. Comparison of site-specific rate-inference methods for protein sequences: empirical Bayesian methods are superior. Mol Biol Evol. 2004;21(9):1781-91. Epub 2004/06/18. doi: 10.1093/molbev/msh194. PubMed PMID: 15201400.

77. Pupko T, Bell RE, Mayrose I, Glaser F, Ben-Tal N. Rate4Site: an algorithmic tool for the identification of functional regions in proteins by surface mapping of evolutionary determinants within their homologues. Bioinformatics. 2002;18 Suppl 1:S71-7. Epub 2002/08/10. PubMed PMID: 12169533.

78. Altschul SF, Gish W. Local alignment statistics. Methods Enzymol. 1996;266:460-80. Epub 1996/01/01. PubMed PMID: 8743700.

79. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol. 1999;292(2):195-202. Epub 1999/09/24. doi: 10.1006/jmbi.1999.3091. PubMed PMID: 10493868.

80. Sigrist CJ, de Castro E, Cerutti L, Cuche BA, Hulo N, Bridge A, Bougueleret L, Xenarios I. New and continuing developments at PROSITE. Nucleic acids research. 2013;41(Database issue):D344-7. Epub 2012/11/20. doi: 10.1093/nar/gks1067. PubMed PMID: 23161676; PMCID: 3531220.

81. Salama RA, Stekel DJ. A non-independent energy-based multiple sequence alignment improves prediction of transcription factor binding sites. Bioinformatics. 2013;29(21):2699-704. doi: 10.1093/bioinformatics/btt463. PubMed PMID: 23990411.

82. Waterhouse AM, Procter JB, Martin DM, Clamp M, Barton GJ. Jalview Version 2--a multiple sequence alignment editor and analysis workbench. Bioinformatics. 2009;25(9):1189-91. doi: 10.1093/bioinformatics/btp033. PubMed PMID: 19151095; PMCID: 2672624.

83. Johnson JM, Mason K, Moallemi C, Xi H, Somaroo S, Huang ES. Protein family annotation in a multiple alignment viewer. Bioinformatics. 2003;19(4):544-5. PubMed PMID: 12611813.

84. Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ, Higgins DG. Clustal W and Clustal X version 2.0. Bioinformatics. 2007;23(21):2947-8. doi: 10.1093/bioinformatics/btm404. PubMed PMID: 17846036.

85. Martin AC. Viewing multiple sequence alignments with the JavaScript Sequence Alignment Viewer (JSAV). F1000Res. 2014;3:249. doi: 10.12688/f1000research.5486.1. PubMed PMID: 25653836; PMCID: 4304231.

86. Schneider TD, Stephens RM. Sequence logos: a new way to display consensus sequences. Nucleic Acids Res. 1990;18(20):6097-100. PubMed PMID: 2172928; PMCID: 332411.

87. Goecks J, Nekrutenko A, Taylor J, Galaxy T. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome Biol. 2010;11(8):R86. doi: 10.1186/gb-2010-11-8-r86. PubMed PMID: 20738864; PMCID: 2945788.

88. Blankenberg D, Von Kuster G, Coraor N, Ananda G, Lazarus R, Mangan M, Nekrutenko A, Taylor J. Galaxy: a web-based genome analysis tool for experimentalists. Current protocols in molecular biology / edited by Frederick M Ausubel [et al].

2010;Chapter 19:Unit 19 0 1-21. doi: 10.1002/0471142727.mb1910s89. PubMed PMID: 20069535; PMCID: 4264107.

89. Giardine B, Riemer C, Hardison RC, Burhans R, Elnitski L, Shah P, Zhang Y, Blankenberg D, Albert I, Taylor J, Miller W, Kent WJ, Nekrutenko A. Galaxy: a platform for interactive large-scale genome analysis. Genome research. 2005;15(10):1451-5. doi: 10.1101/gr.4086505. PubMed PMID: 16169926; PMCID: 1240089.

90. Ramirez-Gonzalez RH, Uauy C, Caccamo M. PolyMarker: A fast polyploid primer design pipeline. Bioinformatics. 2015. doi: 10.1093/bioinformatics/btv069. PubMed PMID: 25649618.

91. Ramirez-Gonzalez RH, Segovia V, Bird N, Fenwick P, Holdgate S, Berry S, Jack P, Caccamo M, Uauy C. RNA-Seq bulked segregant analysis enables the identification of high-resolution genetic markers for breeding in hexaploid wheat. Plant biotechnology journal. 2014. doi: 10.1111/pbi.12281. PubMed PMID: 25382230.

92. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL. GenBank. Nucleic Acids Res. 2005;33(Database issue):D34-8. doi: 10.1093/nar/gki063. PubMed PMID: 15608212; PMCID: 540017.

93. Kozlowski LP, Bujnicki JM. MetaDisorder: a meta-server for the prediction of intrinsic disorder in proteins. BMC Bioinformatics. 2012;13:111. doi: 10.1186/1471-2105-13-111. PubMed PMID: 22624656; PMCID: 3465245.

94. Sievers F, Wilm A, Dineen D, Gibson TJ, Karplus K, Li W, Lopez R, McWilliam H, Remmert M, Soding J, Thompson JD, Higgins DG. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. Molecular systems biology. 2011;7:539. doi: 10.1038/msb.2011.75. PubMed PMID: 21988835; PMCID: 3261699.

95. Nikoopour E, Singh B. Reciprocity in microbiome and immune system interactions and its implications in disease and health. Inflammation & allergy drug targets. 2014;13(2):94-104. PubMed PMID: 24678760.

96. Qin J, Li Y, Cai Z, Li S, Zhu J, Zhang F, Liang S, Zhang W, Guan Y, Shen D, Peng Y, Zhang D, Jie Z, Wu W, Qin Y, Xue W, Li J, Han L, Lu D, Wu P, Dai Y, Sun X, Li Z, Tang A, Zhong S, Li X, Chen W, Xu R, Wang M, Feng Q, Gong M, Yu J, Zhang Y, Zhang M, Hansen T, Sanchez G, Raes J, Falony G, Okuda S, Almeida M, LeChatelier E, Renault P, Pons N, Batto JM, Zhang Z, Chen H, Yang R, Zheng W, Li S, Yang H, Wang J, Ehrlich SD, Nielsen R, Pedersen O, Kristiansen K, Wang J. A metagenome-wide association study of gut microbiota in type 2 diabetes. Nature. 2012;490(7418):55-60. doi: 10.1038/nature11450. PubMed PMID: 23023125.

97. Kolmeder CA, de Vos WM. Metaproteomics of our microbiome - developing insight in function and activity in man and model systems. Journal of proteomics. 2014;97:3-16. doi: 10.1016/j.jprot.2013.05.018. PubMed PMID: 23707234.

98. Rondon MR, August PR, Bettermann AD, Brady SF, Grossman TH, Liles MR, Loiacono KA, Lynch BA, MacNeil IA, Minor C, Tiong CL, Gilman M, Osburne MS, Clardy J, Handelsman J, Goodman RM. Cloning the soil metagenome: a strategy for accessing the genetic and functional diversity of uncultured microorganisms. Applied and environmental microbiology. 2000;66(6):2541-7. PubMed PMID: 10831436; PMCID: 110579.

99. Caporaso JG, Kuczynski J, Stombaugh J, Bittinger K, Bushman FD, Costello EK, Fierer N, Pena AG, Goodrich JK, Gordon JI, Huttley GA, Kelley ST, Knights D, Koenig JE, Ley RE, Lozupone CA, McDonald D, Muegge BD, Pirrung M, Reeder J, Sevinsky JR, Turnbaugh PJ, Walters WA, Widmann J, Yatsunenko T, Zaneveld J, Knight R. QIIME allows analysis of high-throughput community sequencing data. Nat Methods. 2010;7(5):335-6. doi: 10.1038/nmeth.f.303. PubMed PMID: 20383131; PMCID: 3156573.

100. Markowitz VM, Chen IM, Palaniappan K, Chu K, Szeto E, Grechkin Y, Ratner A, Jacob B, Huang J, Williams P, Huntemann M, Anderson I, Mavromatis K, Ivanova NN, Kyrpides NC. IMG: the Integrated Microbial Genomes database and comparative analysis system. Nucleic Acids Res. 2012;40(Database issue):D115-22. doi: 10.1093/nar/gkr1044. PubMed PMID: 22194640; PMCID: 3245086.

101. Meyer F, Paarmann D, D'Souza M, Olson R, Glass EM, Kubal M, Paczian T, Rodriguez A, Stevens R, Wilke A, Wilkening J, Edwards RA. The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes. BMC Bioinformatics. 2008;9:386. doi: 10.1186/1471-2105-9-386. PubMed PMID: 18803844; PMCID: 2563014.

102. Kodama Y, Shumway M, Leinonen R, International Nucleotide Sequence Database C. The Sequence Read Archive: explosive growth of sequencing data. Nucleic Acids Res. 2012;40(Database issue):D54-6. doi: 10.1093/nar/gkr854. PubMed PMID: 22009675; PMCID: 3245110.

103. Huang da W, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. Nature protocols. 2009;4(1):44-57. doi: 10.1038/nprot.2008.211. PubMed PMID: 19131956.

104. Franceschini A, Szklarczyk D, Frankild S, Kuhn M, Simonovic M, Roth A, Lin J, Minguez P, Bork P, von Mering C, Jensen LJ. STRING v9.1: protein-protein interaction networks, with increased coverage and integration. Nucleic Acids Res.

2013;41(Database issue):D808-15. doi: 10.1093/nar/gks1094. PubMed PMID: 23203871; PMCID: 3531103.

105. Xu D, Jaroszewski L, Li Z, Godzik A. FFAS-3D: improving fold recognition by including optimized structural features and template re-ranking. Bioinformatics. 2014;30(5):660-7. doi: 10.1093/bioinformatics/btt578. PubMed PMID: 24130308; PMCID: 3933871.

106. Daniel H, Moghaddas Gholami A, Berry D, Desmarchelier C, Hahne H, Loh G, Mondot S, Lepage P, Rothballer M, Walker A, Bohm C, Wenning M, Wagner M, Blaut M, Schmitt-Kopplin P, Kuster B, Haller D, Clavel T. High-fat diet alters gut microbiota physiology in mice. The ISME journal. 2014;8(2):295-308. doi: 10.1038/ismej.2013.155. PubMed PMID: 24030595; PMCID: 3906816.

107. Perkins DN, Pappin DJ, Creasy DM, Cottrell JS. Probability-based protein identification by searching sequence databases using mass spectrometry data. Electrophoresis. 1999;20(18):3551-67. doi: 10.1002/(SICI)1522-2683(19991201)20:18<3551::AID-ELPS3551>3.0.CO;2-2. PubMed PMID: 10612281.

108. Zybailov B, Mosley AL, Sardiu ME, Coleman MK, Florens L, Washburn MP. Statistical analysis of membrane proteome expression changes in Saccharomyces cerevisiae. Journal of proteome research. 2006;5(9):2339-47. doi: 10.1021/pr060161n. PubMed PMID: 16944946.

109. Pruitt KD, Brown GR, Hiatt SM, Thibaud-Nissen F, Astashyn A, Ermolaeva O, Farrell CM, Hart J, Landrum MJ, McGarvey KM, Murphy MR, O'Leary NA, Pujar S, Rajput B, Rangwala SH, Riddick LD, Shkeda A, Sun H, Tamez P, Tully RE, Wallin C, Webb D, Weber J, Wu W, DiCuccio M, Kitts P, Maglott DR, Murphy TD, Ostell JM. RefSeq: an update on mammalian reference sequences. Nucleic Acids Res. 2014;42(Database issue):D756-63. doi: 10.1093/nar/gkt1114. PubMed PMID: 24259432; PMCID: 3965018.

110. Barker WC, Garavelli JS, Huang H, McGarvey PB, Orcutt BC, Srinivasarao GY, Xiao C, Yeh LS, Ledley RS, Janda JF, Pfeiffer F, Mewes HW, Tsugita A, Wu C. The protein information resource (PIR). Nucleic Acids Res. 2000;28(1):41-4. PubMed PMID: 10592177; PMCID: 102418.

111. Yachdav G, Kloppmann E, Kajan L, Hecht M, Goldberg T, Hamp T, Honigschmid P, Schafferhans A, Roos M, Bernhofer M, Richter L, Ashkenazy H, Punta M, Schlessinger A, Bromberg Y, Schneider R, Vriend G, Sander C, Ben-Tal N, Rost B. PredictProtein--an open resource for online prediction of protein structural and functional features. Nucleic Acids Res. 2014;42(Web Server issue):W337-43. doi: 10.1093/nar/gku366. PubMed PMID: 24799431.

112. Schlessinger A, Schaefer C, Vicedo E, Schmidberger M, Punta M, Rost B. Protein disorder--a breakthrough invention of evolution? Curr Opin Struct Biol. 2011;21(3):412-8. doi: 10.1016/j.sbi.2011.03.014. PubMed PMID: 21514145.

113. Huang da W, Sherman BT, Lempicki RA. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. Nucleic Acids Res. 2009;37(1):1-13. doi: 10.1093/nar/gkn923. PubMed PMID: 19033363; PMCID: 2615629.

114. Maere S, Heymans K, Kuiper M. BiNGO: a Cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks. Bioinformatics. 2005;21(16):3448-9. doi: 10.1093/bioinformatics/bti551. PubMed PMID: 15972284.

115. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome research. 2003;13(11):2498-504. doi: 10.1101/gr.1239303. PubMed PMID: 14597658; PMCID: 403769.

116. Li D, Dye TD. Power and stability properties of resampling-based multiple testing procedures with applications to gene oncology studies. Computational and mathematical methods in medicine. 2013;2013:610297. doi: 10.1155/2013/610297. PubMed PMID: 24348741; PMCID: 3853148.

117. Benjamini YY, Daniel. . The control of the false discovery rate in multiple testing under dependency. . Ann Statist

. 2001;no. 4,: 1165--88. doi: doi:10.1214/aos/1013699998.

118. Kristensen DM, Kannan L, Coleman MK, Wolf YI, Sorokin A, Koonin EV, Mushegian A. A low-polynomial algorithm for assembling clusters of orthologous groups from intergenomic symmetric best matches. Bioinformatics. 2010;26(12):1481-7. doi: 10.1093/bioinformatics/btq229. PubMed PMID: 20439257; PMCID: 2881409.

APPENDIX A – LIST OF FIGURES

APPENDIX B – LIST OF TABLES

APPENDIX C – SUPPORTING MATERIAL

List S1 - list of genomes included in the gold standard reference dataset:

Akkermansia muciniphila ATCC BAA-835, Alistipes putredinis DSM 17216, Alistipes shahii, Bacteroides caccae ATCC 43185, Bacteroides dorei DSM 17855, Bacteroides finegoldii DSM 17565, Bacteroides fragilis NCTC 9343, Bacteroides intestinalis DSM 17393, Bacteroides ovatus ATCC 8483, Bacteroides stercoris ATCC 43183, Bacteroides thetaiotaomicron VPI-5482, Bacteroides uni- formis, Bacteroides vulgatus ATCC 8482, Bacteroides xylanisolvens XB1A, Bifidobacterium longum subsp. infantis ATCC 15697, Blautia hansenii DSM 20583, Blautia wexlerae, Clostridium asparagi- forme DSM 15981, Clostridium clostridioforme CM201, Clostridium leptum DSM 753, Clostridium nexile DSM 1787, Clostridium scindens ATCC 35704, Collinsella aerofaciens ATCC 25986, Copro- coccus comes ATCC 27758, Dorea formicigenerans ATCC 27755, Dorea longicatena DSM 13814, Enterococcus faecalis V583, Eubacterium eligens ATCC 27750, Eubacterium hallii DSM 3353, Eubac- terium ramulus ATCC 29099, Eubacterium rectale ATCC 33656, Faecalibacterium prausnitzii L2-6, Flavonifractor plautii ATCC 29863, Parabacteroides distasonis ATCC 8503, Parabacteroides merdae ATCC 43184, Roseburia intestinalis M50-1, Roseburia inulinivorans DSM 16841, Ruminococcus bromii L2-63, Ruminococcus gnavus ATCC 29149, Ruminococcus obeum

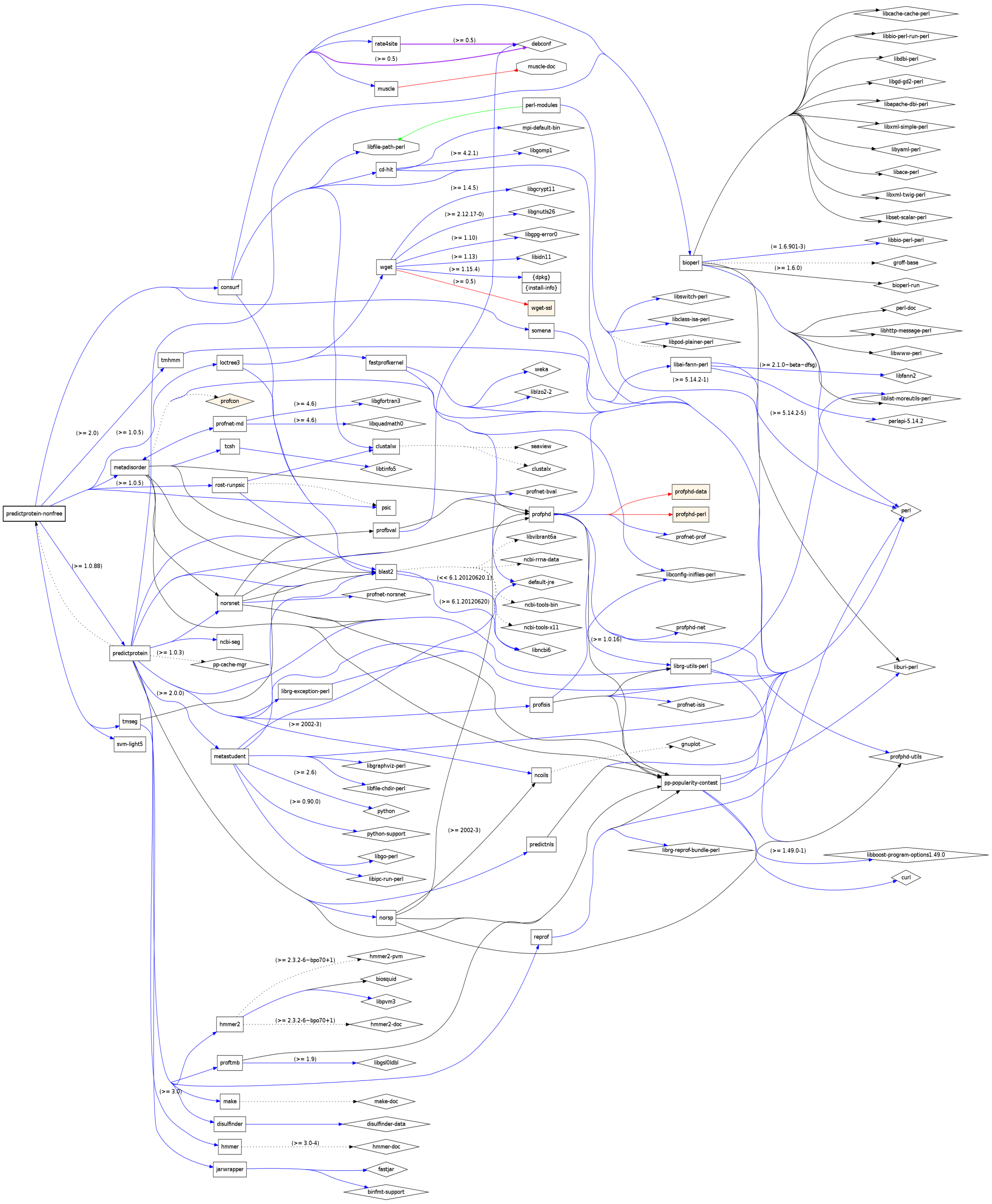


Figure S2 – dependency graph for the package predictprotein-nonfree (most inclusive version of predictprotein). The graph was created with the debtree(1) utility with the following parameters --with-suggests (includes suggested packages) –condense (reduce graph density) --max-depth=3 (limits the number of traversed levels to 3). Blue arrows show a regular dependency relation, black arrows a recommends relation, dotted black suggested package, green inverted arrow denotes a provides relation, red denotes conflict, purple arrows denote a pre-depend relationship. Boxes represent packages, diamond represent end packages. In case such exist, version requirement are labeled on arrows.

Table S3 - Methods incorporated into PredictProtein

Category	Method/dB Name	Description	Command line tool		Output available in		
			Name	Version	Visual	HTML	TEXT
Database search	BLAST	Pairwise alignment	blastall	2.2.26	-	-	x
Database search	PSI-BLAST	Profile based alignment	blastpgp	2.2.27	x	x	x
Database search	HMMER	Hidden markov model search	hmm2pfam	2.3.2	-	-	x
Database search	ps_scan	PROSITE scanning program	ps_scan.pl	1.67	-	x	x
Database Search	PSSH	Mapping of sequence positions onto a structure	generate_pssh2	1.0.0	-	-	x
Database Search	Species Mapper	Maps organism code to kingdom	speciesmapper	1.0,1	x	-	-
Database Search	ID Mapper	maps ids across major databases	idmapper	1.0.3	x	-	-
Analysis method	SEG	Low complexity regions markup	ncbi-seg	0.0.20000620	-	x	x
Analysis method	NCOILS	Calculates the probability that the sequence will adopt a coiled-coil conformation	ncoils	2002	-	x	x
Analysis method	HSSP	Homology derived secondary Structure of proteins	hssp_filter	1	-	-	x
Prediction method	NORSp	NOn-Regular Secondary Structure	norsp	1.0.3	-	x	x
PHDhtm	PHDhtm	Prediction of membrane helices	phd.pl	1.0.40	x	x	x
Prediction method	TMSEG	Prediction of membrane helices	tmseg	1.0.0	x	x	x

Table continues on the following page

Category	Method/dB Name	Description	Command line tool	Output available in	x	x	x
			Name	Version	Visual	HTML	TEXT
Prediction method	PROFsec	Prediction of secondary structure state	prof	1.0.40	x	x	x
Prediction method	PROFacc	Prediction of solvent accessbility	prof	1.0.40	x	x	x
Prediction method	Reprof	Improved prediction of secondary structure state	reprof	1.0.0	-	-	x
Prediction method	PROFBval	Prediction of residue mobility	profbval	1.0.16	x	x	x
Prediction method	NorsNet	Prediction protein disordered sites	norsnet	1.0.16	x	x	x
Prediction method	UCON	Contact based prediction of disordered sites	ucon	1.0.8	x	x	x
Prediction method	Meta-Disorder	Consensus based prediction of protein disorder	metadisorder	1.0.14	x	x	x
Prediction method	ISIS2	Prediction of protein-protein interaction sites	profisis2	1.0.0	x	x	x
Prediction method	SomeNA	Prediction of protein –DNA, -RNA binding sites	somena	1.0.0	X	-	-
Prediction method	LocTree3	Prediction of sub-cellular localization for all domains of life	loctree3	1.0.5	x	x	x
Prediction method	PredictNLS	Prediction of Nuclear Localization Signals (NLS)	predictnls	1.0.18	-	x	x
Prediction method	metastudent	Prediction of GO terms for Molecular Function and Biological Process	metastudent	1.0.9	x	-	X
Prediction method	SNAP2	Prediction of functional changes due to single nucleotide polymorphism	snap2	1.0.10	x	-	x
Prediction method	ConSurf	Identification of functional sites	consurf	1.0.0	x	-	x
Database	UniRef	Clustered set of sequences	N/A	Updates monthly	x	x	X

Table continues on the following page

Category	Method/dB Name	Description	Command line tool	Output available in	x	x	x
			Name	Version	Visual	HTML	TEXT
Database	BIG	non-redundant combination of Swiss-Prot, TrEMBL, PDB	N/A	Updates monthly	x	x	x
Database	PDB	Repository of protein structures	N/A	Updates monthly	x	x	x
Database	Pfam-A	Protein families	N/A	Updates quarterly	x	x	x
Database	PROSITE	Database of biologically significant sites, patterns and profiles	N/A	Updates quarterly	x	x	X

APPENDIX D – PEER REVIEWED PUBLICATIONS

The manuscripts of the following peer-reviewed publications have been appended:

- **Yachdav G**, Kloppmann E, Kajan L, Hecht M, Goldberg T, Hamp T, Honigschmid P, Schafferhans A, Roos M, Bernhofer M, Richter L, Ashkenazy H, Punta M, Schlessinger A, Bromberg Y, Schneider R, Vriend G, Sander C, Ben-Tal N, Rost B. *PredictProtein--an open resource for online prediction of protein structural and functional features*. Nucleic Acids Res. 2014;42(Web Server issue):W337-43.

- **Yachdav G**, Hecht M, Pasmanik-Chor M, Yeheskel A, Rost B. *HeatMapViewer: interactive display of 2D data in biology*. F1000Res. 2014;3:48.

- **Yachdav G**, Kajan L, Vicedo E, Steinegger M, Mirdita M, Angermuller C, Bohm A, Domke S, Ertl J, Mertes C, Reisinger E, Staniewski C, Rost B. *Cloud prediction of protein structure and function with PredictProtein for Debian*. Biomed Res Int. 2013;2013:398968.

Summaries of the publications and my individual contributions are as follows:

Cloud prediction of protein structure and function with PredictProtein for Debian

Bioinformatics software is playing a pivotal role in processing and analyzing biological data. Many groups are making their methods and databases available to the research community through various means including the creation of stand alone downloadable software packages and online services. For over two decades our group has been researching, developing and contributing software for the prediction of protein structure and function features. We have integrated many of those methods into the PredictProtein protein structure and function prediction suite.

Yet the increase in size and scope of PredictProtein created multiple challenges for the medium and long-term maintainability of each method and the software suite in general. Issues such as software resilience, scalability, extendibility and fault tolerance required us to redesign and redevelop the software we provide so we will be able to continue to support the tens of thousands of researchers that use our computational resources each year.

To that end we applied a set of software engineering best practices that standardized our code-base and enabled the creation of a new scalable and extendable annotation pipeline. Furthermore we went to great lengths to make our software available in open source form through established and well-known operating systems. We recoded and packaged nearly ninety bioinformatics tools and databases along with supporting libraries, parsers and utilities and made those available for Debian Linux and derivative operating systems, such as Ubuntu. As scalable, cost effective cloud computing is increasingly being used by the research community, we also made a readily deployable cloud version of PredictProtein.

In this publication we discussed the need for readily available bioinformatics software in handling data generated by "next generation" sequencing (NGS) projects. We detailed our methodology in creating reliable software tools and making them available as packages and machine images that can used in the cloud. We then demonstrated the utility of the cloud images in the context of two high throughput studies.

The project design, methodology and implementation were conceived by both myself and Dr. Laszlo Kajan. The reprogramming, packaging and documentation of nearly half of the 89 packages that were created in this project was done by myself. The manuscript was drafted by myself, Laszlo Kajan and Burkhard Rost.

PredictProtein--an open resource for online prediction of protein structural and functional features

The annotation gap – the difference between the rate of growth of known sequene and expertly annotated sequence is growing daily. At the end of 2014 less than 1% of UniProt, the largest database of known protein sequences, contains expert annotation sequences. Computational tools have been increasing in acccuracy and quality and can now help in annotating proteins for which very little is known.

PredictProtein is an annotation server that has been operated by our group for over twenty years. Over the past decade more than 650 thousands researchers used PredictProtein to annotate 13 million proteins. The server returns multiple sequence alignment, functional motifs and Pfam domains (25), and annotates protein structure and function features. Additionally the service can be used to preform mutability landscape studies (48) -- predicting the effect of all possible nsSNPs.

In this publication we reported the release of a new and improved version of the PredictProtein online service. We first discuss the inclusion of a set of new methods: TMSEG (41) (transmembrane helices prediction), SNAP2 (46) (effect of single amino acid substitution on protein function), LocTree3 (50) (subcellular localization prediction), metastudent (53) (prediction of GO terms), Meta-disorder (65) (prediction or protein disorder), ISIS (40) (prediction of protein-protein binding sites), SomeNA (71) (protein DNA/RNA binding sites) , ConSurf (72, 73) (estimation of evolutionary rates in protein families).

In this project we constructd visualPP – a web applicatoin that unifies the presentation of multiple preditions into a single interactive overview image. Depending on the protein, the overview features may include predictions of secondary structure and solvent accessibility, transmembrane helices, disulfide bonds and disordered regions. Other views in visualPP present additional annotations and predictions, e.g. functional landscapes of the effect of point mutations (SNAP2), predicted GO terms (metastudent) or subcellular localization (LocTree3). In the dashboard viewer, users can mouse over the different view landmarks to reveal more information on the annotations. Additional new features include the PredictProtien cache, a reposiotry of over 15 million pre-calcualted PredcitProtein entries used to speed up the response time of the service. Fianly we report a use case demonstrating the usefulness of the service.

The redevelopment project of PredictProtein was conceived by myself and Burkhard Rost. The prorgramming of the PredictProtein web application, and visualPP was carried out by me. The programming of the PredictProtein annotation pipeline was performed by me and Laszlo Kajan. The manuscript was drafted by me, Edda Kloppman and Burkhard Rost. Edda kloppman provided the use case presented in the manuscript.

HeatMapViewer: interactive display of 2D data in biology

Biological data are often organized into matrices in which the rows signify different items of interest (a gene, a subject, a probe or a position in a sequence), while the columns describe different experiments, variations, or samples. Matrices are easy to process by algorithms. In contrast, the details in large matrices are often, at best, challenging for experts who want to "understand" the data. The information in matrices is usually better digested if presented by 3D plots or heat maps. Heat maps are essentially simplified versions of 3D plots that replace the 3rd dimension with color gradients, thereby conveniently displaying the information contained in matrices. Such heat maps allow for easy visual differentiation between high and low values in a matrix.

Such heat maps are, for example, commonly used to display microarray data as they quickly show which genes (rows) are differentially expressed under some conditions (columns). Microarray technologies utilize arrays of probes located on different exons for each gene and can be helpful in determining gene function by measuring transcription and translation levels under certain experimental conditions. The expression values for the differential expression may be presented at the exon level, correlated with protein domains, and may help to decipher a complex gene expression pattern.

In this publication we report the development of a new JavaScript based HeatMapViewer component that visualize 3D plot on the web. The component uses renders scalable vector graphics (SVG) images that are native to all modern web browsers. The HeatMapViewer features an overview and zoom-in panels; a user control is provided in the form of a frame that can be dragged along the main heat map to determine which area of the heat map should appear in the zoom-in panel. Additionally, a scale bar is presented to show the value ranges and which colors correspond to those values. Finally, each cell in the heat map is associated with a mouse-over event that pops-up tooltips showing the data-value of the cell.

The publication presents two use cases that demonstrate the utility of the HeatMapViewer: 1) microarray expression data from a Retinitis Pigmentosa (RP) study is visualized and analyzed using the component. 2) a mutability landscape analysis of the 7TM human rhodopsin protein is performed and nsSNP predictions are presented using the HeatMapViewer.

The development of the HeatMapViewer was conceived by myself and the implimentation was carried out by me. I also created a web server that lets users visualize HeatMaps online. The manuscript was drafted by myself and Burkhard Rost. Maximillian Hecht and Metsada Pasmanik-Chor provided the use cases. Maximillian Hecht provided Figure 2: The HeatMapViewer component displays the mutability landscape of OPSD_HUMAN.

Hindawi Publishing Corporation
BioMed Research International
Volume 2013, Article ID 398968, 6 pages
http://dx.doi.org/10.1155/2013/398968

Research Article

Cloud Prediction of Protein Structure and Function with PredictProtein for Debian

László Kaján,[1] Guy Yachdav,[1,2,3] Esmeralda Vicedo,[1] Martin Steinegger,[1] Milot Mirdita,[1] Christof Angermüller,[1] Ariane Böhm,[1] Simon Domke,[1] Julia Ertl,[1] Christian Mertes,[1] Eva Reisinger,[1] Cedric Staniewski,[1] and Burkhard Rost[1,2,3,4,5]

[1] *TUM, Department of Informatics, Bioinformatics & Computational Biology-I12, Boltzmannstraß 3, 85748 Garching, Germany*
[2] *Columbia University, Department of Biochemistry and Molecular Biophysics and New York Consortium on Membrane Protein Structure (NYCOMPS), 701 West 168th Street, New York, NY 10032, USA*
[3] *Biosof LLC, 10th Floor, 138 West 25th Street, New York, NY 10001, USA*
[4] *WZW-Weihenstephan, Alte Akademie 8, Freising, Germany*
[5] *Institute for Advanced Study (TUM-IAS), Lichtenbergstraß 2a, 85748 Garching, Germany*

Correspondence should be addressed to László Kaján; lkajan@rostlab.org

Received 4 January 2013; Accepted 5 July 2013

Academic Editor: Ching-Hsien (Robert) Hsu

We report the release of PredictProtein for the Debian operating system and derivatives, such as Ubuntu, Bio-Linux, and Cloud BioLinux. The PredictProtein suite is available as a standard set of open source Debian packages. The release covers the most popular prediction methods from the Rost Lab, including methods for the prediction of secondary structure and solvent accessibility (profphd), nuclear localization signals (predictnls), and intrinsically disordered regions (norsnet). We also present two case studies that successfully utilize PredictProtein packages for high performance computing in the cloud: the first analyzes protein disorder for whole organisms, and the second analyzes the effect of all possible single sequence variants in protein coding regions of the human genome.

1. Background

Bioinformatics is embracing cloud computing. Recent months have seen the publication of cloud sequence analysis platforms, CloVR [1] and Galaxy Cloud [2], and the cloud version of Bio-Linux [3], Cloud BioLinux [4]. Cost analysis depicts cloud computing as an attractive and sustainable solution for computational biology and bioinformatics [5–8]. The rate of data generation of "next generation" sequencing (NGS) drives the efforts to turn to cloud computing as a solution to handling peak-time loads, without the need to maintain large clusters [9]. Cloud-enabled bioinformatics tools are now available in the context of high throughput sequencing and genomics [10].

The Rost Lab provides protein structure and function prediction tools for cloud computing in the PredictProtein suite [11]. PredictProtein began as an Internet server for sequence analysis and the prediction of aspects of protein structure and function in 1992 [12]. Queried with a protein sequence, PredictProtein returns secondary structure and accessibility predictions, predictions of unstructured loops, nuclear localization signals, protein-protein interaction sites, disulfide bonds, regions lacking regular secondary structure, protein family hits, low-complexity regions, bacterial transmembrane beta barrels, coiled-coil regions, protein residue flexibility, and homologous sequences (Figure 1).

Cloud computing is commonly realized on machine instances that run on virtual hardware providing "infrastructure as a service" (IaaS) [13, 14]. This type of cloud computing instantiates compute nodes from machine images. Machine images usually contain an operating system with software tools. For example, one could request the instantiation of 10 worker nodes of PredictProtein on Debian operating system at the Amazon EC2 IaaS offering.

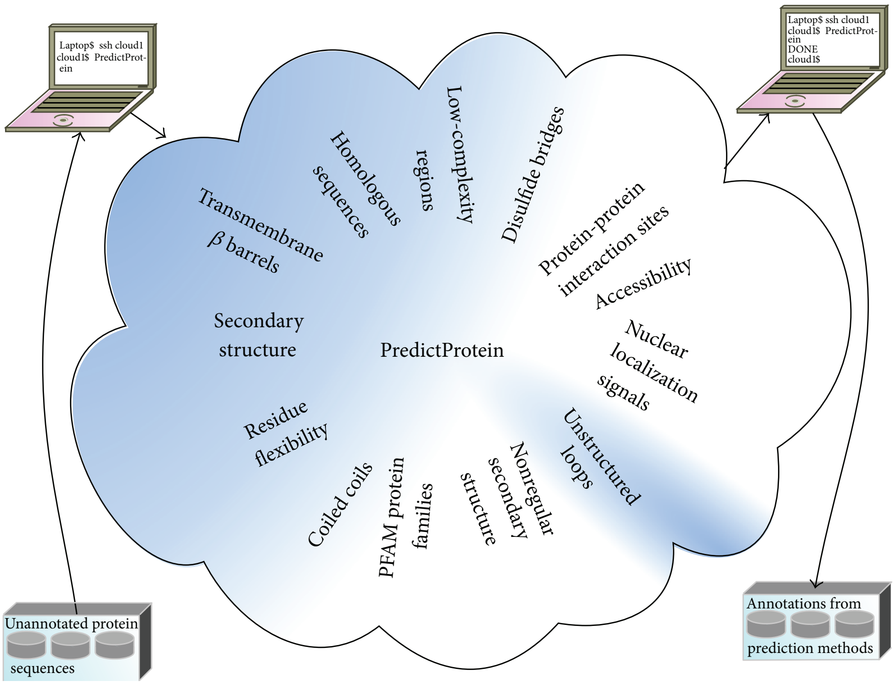


FIGURE 1: Protein annotation by PredictProtein. PredictProtein annotates input sequences with the features shown.

The PredictProtein cloud solution builds upon the open source operating system Debian [15] and provides its functionality as a set of free [16] software packages. Bio-Linux is an operating system for bioinformatics and computational biology. The latest Bio-Linux release 7 provides more than 500 bioinformatics programs on an Ubuntu Linux base [17]. Ubuntu is a "derivative" operating system [18] based on Debian, with its own additions. Cloud BioLinux is a comprehensive cloud solution that is derived from Bio-Linux and Ubuntu. Debian derivatives can easily share packages between each other. For example, Debian packages are automatically incorporated in Ubuntu [19] and are also usable in Cloud BioLinux (the procedure is described in [4]).

2. Implementation

The PredictProtein suite is implemented as a set of free packages released at http://debian.org/. Software packaging conformed with the Policy Manual [20], and following the recommendations of the Developer's Reference [21].

3. Results and Discussion

High-throughput experiments generate vast amounts of data at an ever-increasing rate; the pace of creating reliable annotations needed to use that data increases much slower. One of the major challenges for computational tools is to narrow the resulting increase in the protein annotation gap [22]. Of the over 35 m (million) sequences in the UniProt Knowledgebase 2013_05 [23], only about 500 k (500 thousand) have explicit experimental annotations in Swiss-Prot [24]. Computational prediction methods, such as those included in PredictProtein, can annotate important features for the remainder and enable us to draw scientific insights. Unfortunately, the task is often intractable for any single desktop computer within reasonable time. Fortunately, cloud computing is now at hand. On-demand servers in the cloud promise to fit computing power to most tasks economically, and without a fair portion of the usual worries of system management: hardware purchasing, recruiting a system manager, high availability issues, and so forth ([13] and the references therein). One problem remains: how to get the often adhoc analysis toolset from the desktop environment into the cloud? Directly addressing this problem, here we report the first Debian package release of the protein feature prediction toolset "PredictProtein," developed at the Rost Lab.

The publication of scientific results has, overall, changed surprisingly little since the Internet exists [25]. Research code is regularly distributed as a "zip" file of the development directory. Often, the only "documentation" distributed along with the code is the published paper accompanied by some "README" file. Software distributed this way often fails outside the laboratory without expert attention. In order to address this issue in the PredictProtein suite, we decided to apply the community and time-tested packaging and release requirements of Debian to PredictProtein components. We have traced all dependencies, eliminated convenience copies, carefully documented each of our prediction methods, and made them go through the thorough review process every Debian package receives. This converted PredictProtein from an adhoc implementation to a reusable software component (Figure 2).

Our packages facilitate the generation of purpose-built machine images for cloud computing. As an example, we distribute a slim PredictProtein machine image (PPMI) through

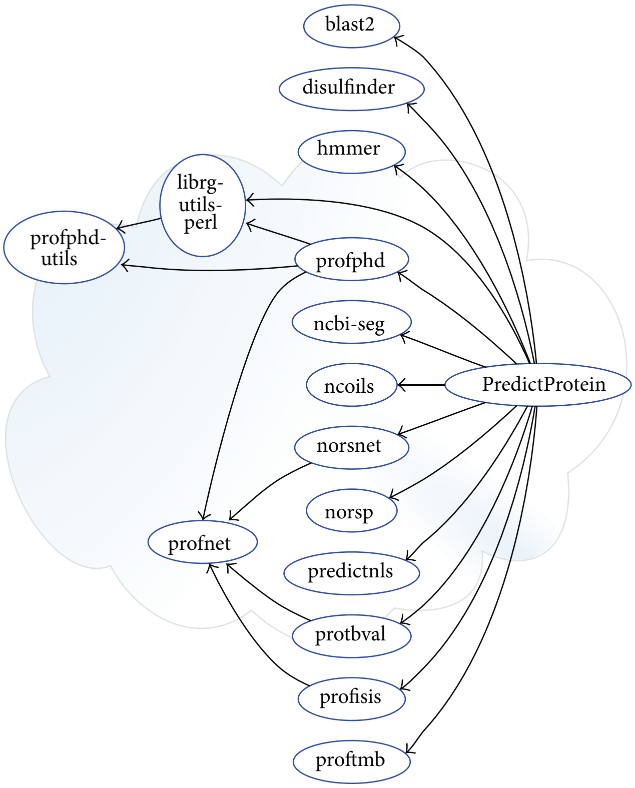


Figure 2: Package dependencies for PredictProtein. Arrows represent "depends on" relationships. Only significant dependencies are shown for clarity. Convenience copies of "profnet" for "profphd," "norsnet," "profbval," and "profisis" have been merged to a single "profnet" package. Similar merging was done for all code convenience copies.

the PredictProtein website [26]. This image contains a minimal installation of Debian with the command line version of PredictProtein. Databases are provided as a separate disk image. The PPMI is bootable on server instances in cloud infrastructure services, or on locally installed virtualization software. The latter allows for a cross-platform solution to use PredictProtein. Apart from virtualization, "chroot" environments present an option to run the software on Linux distributions where Debian packages are not readily usable. After booting the machine image, a friendly message at the login prompt offers usage tips and directions to documentation. A "Getting Started with PredictProtein" guide is available online [27]. The PPMI and the data image are updated regularly and are freely available at http://predictprotein.org/. For a comprehensive bioinformatics and computational biology computing environment, we recommend using PredictProtein with Bio-Linux [3] or Cloud BioLinux [4], where PredictProtein is either preinstalled or is easily installable from package repositories. We plan to release the web-based graphical interface of PredictProtein for these platforms in the near future.

The PredictProtein suite has attracted respectable popularity both online and offline. PredictProtein has been operating continuously since 1992, that is, the dawn of the Internet. Today, over 100,000 online users are registered; over 500 users access the PredictProtein web page every day and 12,000 unique users apply the service every month. Our Media Wiki page presenting an overview of the Rost Lab software packages has been accessed nearly 60,000 times since its launch 36 months ago. Adoption of the PredictProtein packages by the community has also been remarkable. Over 200 packages of the PredictProtein suite are installed from the Debian repository alone, while these and other installations have performed over 57 million protein feature predictions over the past year, not counting our own usage. Out of this, ~30 million were secondary structure and accessibility predictions from the "profphd" method [28].

4. PredictProtein Packages

The following protein feature prediction methods—components of PredictProtein—are available (feature—"package name"): secondary structure, accessibility, and transmembrane helices—"profphd" [29–31]; unstructured loops—"norsnet" [32]; nuclear localization signals—"predictnls" [33]; protein-protein interaction sites—"profisis" [34]; disulfide bridges—"disulfinder" [35]; nonregular secondary structure—"norsp" [36]; PFAM hits—"hmmer" [37, 38]; local complexity—"ncbi-seg" [39]; bacterial transmembrane beta barrels—"proftmb" [40]; coiled-coils—"ncoils" [41]; protein residue flexibility—"profbval" [42]; sequence homologies—"blast2" [43]; protein feature prediction suite—"predictprotein" [11].

These tools are available under a free license through Debian and are automatically incorporated into other Linux distributions such as Ubuntu. An overview of the packages offered for bioinformatics and cloud computing, complete with literature references, is available at Debian Med [44]. PredictProtein is listed in the Biology task.

5. Case Study 1: Protein Disorder in Completely Sequenced Organisms

The goal of this study is to collect evidence for three hypotheses on protein disorder: (1) it is more useful to picture disorder as a distinct phenomenon than as an extreme example of protein flexibility; (2) there are many very different flavors of protein disorder, but it is advantageous to recognize just two main types, namely, *well structured* and *disordered*; (3) nature uses protein disorder as a tool to adapt to different environments [45]. We predicted protein disorder both on an inhouse compute grid and on a compute grid manually setup in the OpenNebula [46] cloud service provided by the CSC Finland [47]. Data and tool (the PPMI) images for grid nodes in the cloud were downloaded from http://predictprotein.org/. The PPMI image was extended with a grid client, and a separate machine instance was used as grid master. PredictProtein for the local grid was installed from the main Debian repository. Required databases (28 GB) were included on a data disk image for cloud machine instances. Input to PredictProtein jobs consisted of protein sequences (in total less than 1 GB). Grid job submissions to the local and the cloud grid were manually adjusted according to available resources. Over 9 million disorder predictions were made over the course of the past few years.

6. Case Study 2: Comprehensive In Silico Mutagenesis of Human Proteome

This project aims at providing information about the functional effect of every possible point mutation in all human proteins, that is, for the replacement of 19∗N amino acids for a protein with N residues. Overall, this generated 300 million human sequence variants (point mutants). The method SNAP [48] predicted the effect of each variant, that is, each "nonsynonymous single nucleotide polymorphisms" (nsSNPs) upon protein function. These predictions are useful for several reasons. First, the study of all possible mutations in human will provide the background against which we can assess the effect of mutations that are actually observed between people. This is crucial for both the advance toward personalized medicine and health and the understanding of human diversity and variation. Second, our computation provides quick "look-up" answers available for all the important variants that are observed and implied in important phenotypes. The only way to cover those lookups is by precomputing all the possible changes. SNAP can take advantage of PredictProtein results for faster processing. With the PredictProtein packages presented here, a solution was built in the form of a public Amazon Machine Image (AMI, ami-3f5f8156) that allows running PredictProtein on the Amazon Elastic Compute Cloud (EC2). We extended an Ubuntu-based StarCluster [49] AMI with PredictProtein and its required databases (28 GB). Because every protein can be computed independently, we formed a grid job out of each protein and used the Grid Engine (GE) to distribute work on the machine instances. We used StarCluster to automate grid setup on the EC2. Because a lot of CPU power was needed, the "Cluster Compute Eight Extra Large Instance" was chosen. This instance type is especially crafted for big data with a lot of CPU power. One instance has 60.5 GB memory, 88 EC2 Compute Units (2x Intel Xeon E5-2670, eight-core-architecture "Sandy Bridge"), and 3370 GB instance storage. The sequence variants were analyzed based on the human reference proteome from the National Center for Biotechnology Information (build 37.3, proteins, 21MB). We processed 29,036 sequences with 16,618,608 residues. This amounted to predicting the functional effect of 315,753,552 individual amino acid changes.

7. Conclusion

The open source release of the PredictProtein protein structure and function prediction suite from the Rost Lab is now available for Debian and derivative operating systems, such as Ubuntu, Bio-Linux, and Cloud BioLinux. The software, due to its standard packaging, is readily deployable in the cloud. Successfully addressing the challenges of cloud computing brings PredictProtein—developed over almost two decades—into the present and the future. In accordance with the Rost Lab open policy [50], and supported by anonymous statistics, PredictProtein is now shared with a wide range of users. We encourage the bioinformatics community to take advantage of our open source software, itself a result of the collaboration of the wider open source software community.

Conflict of Interests

The authors declare that they have no conflict of interests.

Authors' Contributions

L. Kaján and G. Yachdav (equal contributors) have redesigned "predictprotein," performed initial software packaging, and wrote the paper; E. Vicedo, M. Steinegger and M. Mirdita performed case studies, and reviewed the paper; C. Angermüller, A. Böhm, S. Domke, J. Ertl, C. Mertes, E. Reisinger, and C. Staniewski finalized the packaging for Debian; B. Rost provided initial implementation of the "predictprotein" core module and reviewed the paper.

Acknowledgments

This work was supported by the Alexander von Humboldt Foundation, the IAS at the TU in Munich and the National Institutes of Health (R01-LM07329 to G. Yachdav). M. Steinegger and M. Mirdita received funding from Amazon and Cycle Computing for case study 2. The authors thank all those who funded our research. The authors thank researchers who deposited data into publicly available datasets and programmers who provided their work under a free license: our work stands upon their shoulders and would not have been possible without them. The authors wish to thank the Debian project in general and Steffen Möller and Andreas Tille in particular for their tireless support.

References

[1] S. V. Angiuoli, M. Matalka, A. Gussman et al., "CloVR: a virtual machine for automated and portable sequence analysis from the desktop using cloud computing," *BMC Bioinformatics*, vol. 12, p. 356, 2011.

[2] E. Afgan, D. Baker, N. Coraor et al., "Harnessing cloud computing with Galaxy Cloud," *Nature Biotechnology*, vol. 29, no. 11, pp. 972–974, 2011.

[3] D. Field, B. Tiwari, T. Booth et al., "Open software for biologists: from famine to feast," *Nature Biotechnology*, vol. 24, no. 7, pp. 801–803, 2006.

[4] K. Krampis, T. Booth, B. Chapman et al., "Cloud BioLinux: pre-configured and on-demand bioinformatics computing for the genomics community," *BMC Bioinformatics*, vol. 13, p. 42, 2012.

[5] J. T. Dudley, Y. Pouliot, R. Chen, A. A. Morgan, and A. J. Butte, "Translational bioinformatics in the cloud: an affordable alternative," *Genome Medicine*, vol. 2, no. 8, p. 51, 2010.

[6] S. V. Angiuoli, J. R. White, M. Matalka, O. White, and W. F. Fricke, "Resources and costs for microbial sequence analysis evaluated using virtual machines and cloud computing," *PLoS ONE*, vol. 6, no. 10, Article ID e26624, 2011.

[7] P. Kudtarkar, T. F. DeLuca, V. A. Fusaro, P. J. Tonellato, and D. P. Wall, "Cost-effective cloud computing: a case study using the comparative genomics tool, roundup," *Evolutionary Bioinformatics*, vol. 2010, no. 6, pp. 197–203, 2010.

[8] M. Steinegger, *HPC Full in Silico Mutagenesis, in Department of Bioinformatics*, Technical University of Munich, Munich, Germany, 2012.

[9] L. D. Stein, "The case for cloud computing in genome informatics," *Genome Biology*, vol. 11, no. 5, p. 207, 2010.

[10] E. Afgan, D. Baker, N. Coraor, B. Chapman, A. Nekrutenko, and J. Taylor, "Galaxy CloudMan: delivering cloud compute clusters," *BMC Bioinformatics*, vol. 11, supplement 12, p. S4, 2010.

[11] B. Rost and J. Liu, "The PredictProtein server," *Nucleic Acids Research*, vol. 31, no. 13, pp. 3300–3304, 2003.

[12] B. Rost, G. Yachdav, and J. Liu, "The PredictProtein server," *Nucleic Acids Research*, vol. 32, pp. W321–W326, 2004.

[13] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 50–55, 2008.

[14] V. A. Fusaro, P. Patil, E. Gafni, D. P. Wall, and P. J. Tonellato, "Biomedical cloud computing with amazon web services," *PLoS Computational Biology*, vol. 7, no. 8, Article ID e1002147, 2011.

[15] J. J. Amor, G. Robles, J. M. González-Barahona, and I. Herraiz, "From pigs to stripes: a travel through debian," in *Proceedings of the Debian Annual Developers Meeting (DebConf '05)*, Citeseer, 2005.

[16] "The Debian Free Software Guidelines (DFSG)," http://www.debian.org/social_contract#guidelines.

[17] B. T. Dawn Field, T. Booth, S. Houten, D. Swan, N. Bertrand, and M. Thurston, "Bio-Linux 7," http://nebc.nerc.ac.uk/tools/bio-linux/bio-linux-7-info, 2012.

[18] "Debian Derivatives," http://wiki.debian.org/Derivatives.

[19] "NEW packages through Debian," https://wiki.ubuntu.com/UbuntuDevelopment/NewPackages#NEW_packages_through_Debian.

[20] "Debian Policy Manual," http://www.debian.org/doc/debian-policy/, 2012.

[21] "Debian Developer's Reference," http://www.debian.org/doc/manuals/developers-reference/, 2012.

[22] Y. Bromberg, G. Yachdav, Y. Ofran, R. Schneider, and B. Rost, "New in protein structure and function annotation: hotspots, single nucleotide polymorphisms and the "Deep Web"," *Current Opinion in Drug Discovery and Development*, vol. 12, no. 3, pp. 408–419, 2009.

[23] M. Magrane and U. Consortium, "UniProt Knowledgebase: a hub of integrated protein data," *Database*, vol. 2011, p. bar009, 2011.

[24] A. Bairoch, B. Boeckmann, S. Ferro, and E. Gasteiger, "Swiss-Prot: juggling between evolution and stability," *Briefings in Bioinformatics*, vol. 5, no. 1, pp. 39–55, 2004.

[25] R. Gentleman, "Reproducible research: a bioinformatics case study," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, p. 1034, 2005.

[26] "PredictProtein Website," http://predictprotein.org/.

[27] L. Kajan, "Getting Started with PredictProtein," http://wiki.debian.org/DebianMed/PredictProtein, 2013.

[28] B. Rost and C. Sander, "Improved prediction of protein secondary structure by use of sequence profiles and neural networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 90, no. 16, pp. 7558–7562, 1993.

[29] B. Rost and C. Sander, "Combining evolutionary information and neural networks to predict protein secondary structure," *Proteins*, vol. 19, no. 1, pp. 55–72, 1994.

[30] B. Rost and C. Sander, "Conservation and prediction of solvent accessibility in protein families," *Proteins*, vol. 20, no. 3, pp. 216–226, 1994.

[31] B. Rost, R. Casadio, P. Fariselli, and C. Sander, "Transmembrane helices predicted at 95% accuracy," *Protein Science*, vol. 4, no. 3, pp. 521–533, 1995.

[32] A. Schlessinger, J. Liu, and B. Rost, "Natively unstructured loops differ from other loops," *PLoS Computational Biology*, vol. 3, no. 7, p. e140, 2007.

[33] M. Cokol, R. Nair, and B. Rost, "Finding nuclear localization signals," *EMBO Reports*, vol. 1, no. 5, pp. 411–415, 2000.

[34] Y. Ofran and B. Rost, "ISIS: interaction sites identified from sequence," *Bioinformatics*, vol. 23, no. 2, pp. e13–e16, 2007.

[35] A. Ceroni, A. Passerini, A. Vullo, and P. Frasconi, "Disulfind: a disulfide bonding state and cysteine connectivity prediction server," *Nucleic Acids Research*, vol. 34, pp. W177–W181, 2006.

[36] J. Liu and B. Rost, "NORSp: predictions of long regions without regular secondary structure," *Nucleic Acids Research*, vol. 31, no. 13, pp. 3833–3835, 2003.

[37] R. D. Finn, J. Mistry, J. Tate et al., "The Pfam protein families database," *Nucleic Acids Research*, vol. 38, supplement 1, pp. D211–D222, 2010.

[38] S. R. Eddy, "Accelerated profile HMM searches," *PLoS Computational Biology*, vol. 7, no. 10, Article ID e1002195, 2011.

[39] J. C. Wootton and S. Federhen, "Statistics of local complexity in amino acid sequences and sequence databases," *Computers and Chemistry*, vol. 17, no. 2, pp. 149–163, 1993.

[40] H. Bigelow and B. Rost, "PROFtmb: a web server for predicting bacterial transmembrane beta barrel proteins," *Nucleic Acids Research*, vol. 34, pp. W186–W188, 2006.

[41] A. Lupas, "[30] Prediction and analysis of coiled-coil structures," *Methods in Enzymology*, vol. 266, pp. 513–524, 1996.

[42] A. Schlessinger, G. Yachdav, and B. Rost, "PROFbval: predict flexible and rigid residues in proteins," *Bioinformatics*, vol. 22, no. 7, pp. 891–893, 2006.

[43] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology*, vol. 215, no. 3, pp. 403–410, 1990.

[44] S. Möller, H. N. Krabbenhöft, A. Tille et al., "Community-driven computational biology with Debian Linux," *BMC Bioinformatics*, vol. 11, supplement 12, p. S5, 2010.

[45] A. Schlessinger, C. Schaefer, E. Vicedo, M. Schmidberger, M. Punta, and B. Rost, "Protein disorder–a breakthrough invention of evolution?" *Current Opinion in Structural Biology*, vol. 21, no. 3, pp. 412–418, 2011.

[46] R. Moreno-Vozmediano, R. Montero, and I. Llorente, "IaaS cloud architecture: from virtualized data centers to federated cloud infrastructures," *IEEE Computer Society*, vol. 45, no. 12, pp. 65–72, 2012.

[47] T. H. Nyrönen, M. A. Babar, C. E. Cuesta, and J. E. Savolainen, "Delivering ICT infrastructure for biomedical research," in *Proceedings of the WICSA/ECSA 2012 Companion Volume*, pp. 37–44, ACM, Helsinki, Finland, 2012.

[48] Y. Bromberg and B. Rost, "SNAP: predict effect of non-synonymous polymorphisms on function," *Nucleic Acids Research*, vol. 35, no. 11, pp. 3823–3835, 2007.

[49] "StarCluster," http://star.mit.edu/cluster/index.html.

[50] R. H. Lathrop and B. Rost, "ISCB public policy statement on open access to scientific and technical research literature," *Bioinformatics*, vol. 27, no. 3, pp. 291–294, 2011.

Published online 05 May 2014 *Nucleic Acids Research, 2014, Vol. 42, Web Server issue* ***W337–W343***
doi: 10.1093/nar/gku366

PredictProtein—an open resource for online prediction of protein structural and functional features

Guy Yachdav[1,2,3,*], Edda Kloppmann[1,4], Laszlo Kajan[1], Maximilian Hecht[1,3], Tatyana Goldberg[1,3], Tobias Hamp[1], Peter Hönigschmid[5], Andrea Schafferhans[1], Manfred Roos[1], Michael Bernhofer[1], Lothar Richter[1], Haim Ashkenazy[6], Marco Punta[7,8], Avner Schlessinger[9], Yana Bromberg[2,10], Reinhard Schneider[11], Gerrit Vriend[12], Chris Sander[13], Nir Ben-Tal[14] and Burkhard Rost[1,2,4,15,16,17]

[1]Department of Informatics, Bioinformatics & Computational Biology i12, TUM (Technische Universität München), Garching/Munich 85748, Germany, [2]Biosof LLC, New York, NY 10001, USA, [3]TUM Graduate School, Center of Doctoral Studies in Informatics and its Applications (CeDoSIA), TUM (Technische Universität München), Garching/Munich 85748, Germany, [4]New York Consortium on Membrane Protein Structure (NYCOMPS), Columbia University, New York, NY 10032, USA, [5]Department of Genome Oriented Bioinformatics, Technische Universität München, Wissenschaftszentrum Weihenstephan, Freising 85354, Germany, [6]The Department of Cell Research and Immunology, George S. Wise Faculty of Life Sciences, Tel Aviv University, 69978 Tel Aviv, Israel, [7]Wellcome Trust Sanger Institute, Hinxton, Cambridgeshire, CB10 1SA, UK, [8]European Molecular Biology Laboratory, European Bioinformatics Institute, Hinxton, Cambridgeshire, CB10 1SD, UK, [9]Department of Pharmacology and Systems Therapeutics, Icahn School of Medicine at Mount Sinai, One Gustave L. Levy Place, New York, NY 10029, USA, [10]Department of Biochemistry and Microbiology, Rutgers University, New Brunswick, NJ 08901, USA, [11]Luxembourg University & Luxembourg Centre for Systems Biomedicine, 4362 Belval, Luxembourg, [12]CMBI, NCMLS, Radboudumc Nijmegen Medical Centre, 6525 GA Nijmegen, The Netherlands, [13]Computational Biology Program, Memorial Sloan Kettering Cancer Center, New York, 10065 NY, USA, [14]The Department of Biochemistry and Molecular Biology, George S. Wise Faculty of Life Sciences, Tel Aviv University, 69978 Tel Aviv, Israel, [15]Department of Biochemistry and Molecular Biophysics & New York Consortium on Membrane Protein Structure (NYCOMPS), Columbia University, New York, NY 10032, USA, [16]Institute for Advanced Study (TUM-IAS), Garching/Munich 85748, Germany and [17]Institute for Food and Plant Sciences WZW-Weihenstephan, Alte Akademie 8, Freising 85350, Germany

Received February 21, 2014; Revised April 04, 2014; Accepted April 15, 2014

ABSTRACT

PredictProtein is a meta-service for sequence analysis that has been predicting structural and functional features of proteins since 1992. Queried with a protein sequence it returns: multiple sequence alignments, predicted aspects of structure (secondary structure, solvent accessibility, transmembrane helices (TMSEG) and strands, coiled-coil regions, disulfide bonds and disordered regions) and function. The service incorporates analysis methods for the identification of functional regions (ConSurf), homology-based inference of Gene Ontology terms (metastudent), comprehensive subcellular localization prediction (LocTree3), protein–protein binding sites (ISIS2), protein–polynucleotide binding sites (SomeNA) and predictions of the effect of point mutations (non-synonymous SNPs) on protein function (SNAP2). Our goal has always been to develop a system optimized to meet the demands of experimentalists not highly experienced in bioinformatics. To this end, the PredictProtein results are presented as both text and a series of intuitive, interactive and visually appealing figures. The web server and sources are available at http://ppopen.rostlab.org.

INTRODUCTION

Molecular biology is moving into the high-throughput mode as the number of experiments needed to support a single hypothesis is rapidly growing. The line between experimental result and computational analysis is blurring; this also shifts what constitutes a reliable annotation. On top, the vast amount of life science data outpaces computer power. For example, less than 1% of the over 51 million

[*]To whom correspondence should be addressed. Tel: +49 (89) 289-17811; Fax: +49 (89) 289-19414; Email: gyachdav@rostlab.org

sequences in UniProt (February 2014) (1) have some expert annotations in Swiss-Prot. This protein annotation gap widens every day (2). PredictProtein is one of the resources applicable to all proteins that contribute to closing this gap.

The PredictProtein (PP) server is an automatic service that searches up-to-date public sequence databases, creates alignments, and predicts aspects of protein structure and function. In 1992, PredictProtein went online as one of the first Internet servers in molecular biology at the EMBL (Heidelberg, Germany). From 1999 to 2009, the server operated from Columbia University (New York, NY) and in 2009 it moved to the TUM (Munich, Germany). PredictProtein was one of the first services realizing state-of-the-art protein sequence analysis, and the prediction of structural and functional features in a single server. While many outstanding services (3) have expanded on some of those aspects, PredictProtein has remained one of the most comprehensive resources. The thousands of citations to PredictProtein and to our methods demonstrate the server's applicability and acceptance. Since 2009, for example, its website was visited more than one million times by about 80 000 unique visitors per year from 139 countries. Furthermore, over 500 000 sequences were submitted and processed by the service. About half of all submitted sequences were not in UniProt (1) at the time of submission. This suggests that the server's primary utility is in providing annotations for uncharacterized proteins. The following two central principles have guided the evolution of PredictProtein.

(1) *Sustained quality with performance estimates.* The performance of many tools is not sufficiently assessed and/or their performance does not sustain over time. Two decades of Critical Assessment of protein Structure Prediction (CASP)-like experiments (4,5) have demonstrated this repeatedly. PredictProtein went online with a method for the prediction of protein secondary structure (PHD (6)) and 22 years later the performance estimates for that method continue to be valid: a unique achievement.

(2) *Ease of use.* From the beginning we have aspired to make the use of our tools intuitive for all users. Unfortunately, the growth in size and scope continues to challenge the realization of this guiding principle. In 1992, the service provided alignments and secondary structure prediction; in 2014, it includes over 30 complex tools. Creating a unified, natural interface for these tools is challenging. Furthermore, we need to invest more resources to sustain the increasing usage as the data flood surges on. For example, most of our CPU goes into running PSI-BLAST (7). Since 2009, databases grew 10-fold whereas the CPU speed has only tripled, i.e. we need at least three times the number of CPUs we currently have to achieve the same ease in handling each job.

METHODS

PredictProtein incorporates over 30 tools

Supplementary Table S1, Supporting Online Material provides a comprehensive list of all components. *Database searches:* sequences similar to the query are identified by standard, pairwise BLAST (8) and iterated PSI-BLAST (7) searches (9,10) against a non-redundant combination of PDB (11), Swiss-Prot (12) and TrEMBL (1). In addition, functional motifs are taken from PROSITE (13) and domains from Pfam (14). *Prediction of structural features:* predicted aspects of structure include PROFphd secondary structure and solvent accessibility (15,16), PROFtmb transmembrane strands (17), TMSEG transmembrane helices, COILS coiled-coil regions (18), DISULFIND disulfide bonds (19) and SEG low-complexity regions (20). Disordered regions are predicted by a set of tools: UCON (21), NORSnet (22), PROFbval (23,24) and Meta-Disorder (25). *Prediction of functional features:* predicted aspects include ConSurf annotations and visualizations of functionally important sites (26,27), protein mutability landscape analysis showing the effect of point mutations on protein function predicted by SNAP2 (28), Gene Ontology (GO) terms from metastudent (29), LocTree3 predictions of subcellular localization (30), protein–protein interaction sites (ISIS2) and protein–DNA, protein–RNA binding sites (SomeNA). Almost all prediction methods use evolutionary information obtained from PSI-BLAST searches; the more related protein sequences are found and the more divergent those are, the higher the gain in performance (10,15). However, none of the methods (with the exception of metastudent, see below) relies solely on profiles and the prediction without a profile is significantly better than random. For most prediction methods (e.g. LocTree3 and SNAP2) the prediction quality is estimated by a reliability score. In the following, we introduce some of the recent and upcoming additions since 2004 (31) in more detail.

New: TMSEG transmembrane helix predictions

TMSEG (Bernhofer, M. *et al.*, in preparation) predicts alpha-helical transmembrane proteins, the position of transmembrane helices, and membrane topology. The method uses a novel segment-based neural network to refine the final prediction. TMSEG was developed and evaluated on 166 transmembrane proteins extracted from PDBTM (32) and OPM (33), and on 1441 proteins from the SignalP4.1 dataset (34). In our hands, TMSEG appears to complement and improve over the best existing methods (e.g. PolyPhobius (35) and Memsat3 (36)) predicting all membrane helices correctly for about 60% of all proteins. The method correctly identifies 98% of all transmembrane proteins with a false positive rate of less than 2%.

New: SNAP2 predict effect of mutations upon function

SNAP2 predicts the effect of single amino acid substitutions on protein function (37). It improves over its predecessor SNAP (38) by using additional coarse-grained features that better classify samples with unclear evidence. With a two-state accuracy of 83% and an AUC of 0.91, SNAP2 performs on par or better than other state-of-the-art methods on human variants while significantly outperforming these methods for other organisms. SNAP2 is the only available method predicting the effect of point mutations even without alignment information (if fewer than 10 related proteins are found, a specific method is applied with an expected accuracy of ∼70% instead of 83%). For each protein we also predict the entire protein mutability landscape (28,39), i.e. the functional effect of all possible point mutations. The results are displayed in a heatmap representation (40) of functional effects (Figure 1C).

New: LocTree3 subcellular localization for all domains of life

LocTree3 predicts subcellular localization for proteins in all domains of life (30). The method predicts the localization in 18 classes (8 classes for transmembrane and 10 classes for soluble proteins) for eukaryotes, in 6 for bacteria and in 3 for archaea. LocTree3 successfully combines de novo (41) and homology-based predictions (7), reaching an 18-state prediction accuracy over 80% for eukaryotes and a 6-state accuracy over 89% for bacteria. The high level of performance and the large number of predicted classes make LocTree3 the most comprehensive and most accurate tool for subcellular localization prediction.

New: metastudent infers GO terms by homology

The method metastudent (29) predicts GO (42) terms through homology inference. It first BLASTs queries against proteins with experimental GO annotations taken from Swiss-Prot (12), i.e. when no hit to any protein with experimentally annotated GO term is returned, no prediction is made. Then, three algorithms independently choose which GO terms to inherit. These differ in the amount and quality of alignment hits considered and how they assign a probability to each GO term. A meta-classifier combines the three through linear regression. metastudent achieves a maximum F1 score of 0.36 in the biological process ontology and of 0.48 in the molecular function ontology (29). Although this is slightly worse (within the error estimates (43)) than the best method for predicting GO terms (44), the advantage is that metastudent predictions can easily be traced back to the experimental annotations upon which they are based.

Recent: Meta-Disorder prediction of protein disorder

Intrinsically disordered or unstructured regions in proteins do not fold into well-defined three-dimensional (3D) structures when in isolation, but may become structured upon binding to a substrate. Because of the heterogeneity of disordered regions, we have developed several methods predicting different types of disorders. UCON (21) combines protein-specific pairwise contacts predicted by PROFcon (45) with pairwise statistical potentials to predict long disordered regions that are rendered intrinsically unstructured by few internal connections. NORSnet (22) predicts disordered regions with NO Regular Secondary structure (NORS (46), i.e. long loops), separating very long disordered loops predicted by NORSp (47) from all other regions in the PDB (11). PROFbval (23,24), trained on B-values in X-ray structures, predicts flexible residues in short disordered regions. Meta-Disorder (25) is a neural-network-based meta-predictor that uses different sources of information, including the orthogonal disorder predictors mentioned above and others, e.g. IUPred (48) and DISOPRED (49). Meta-Disorder significantly outperforms its constituents (25,50). A comprehensive, independent study (50), on disordered regions from the PDB and DisProt (51), suggested Meta-Disorder to be one of the top two methods available.

Recent: protein–protein binding sites

Residues that can bind other proteins are now predicted by ISIS2 instead of ISIS (52). ISIS splits a query sequence into windows of nine consecutive residues, encoding each window as a vector of features (e.g. PSI-BLAST amino acid conservation frequencies or predicted secondary structure). A neural network, trained on existing protein–protein binding residue annotations, determines whether a query residue can bind other proteins. ISIS2 has been trained on a large dataset of PDB-annotated binding sites (53). A faster neural network implementation (53) and new methods for predicting residue features further improve the accuracy of ISIS2.

Recent: protein–DNA, protein–RNA binding sites

Protein–polynucleotide binding underlies important processes such as replication and transcription. SomeNA (54) predicts protein–polynucleotide binding on three levels. First, it predicts which proteins bind nucleotides. Second, it predicts the type of binding (RNA or DNA or both). Third, it predicts the protein residues that bind DNA or RNA. The first step is performed best: 77% of the proteins are correctly predicted to bind DNA and RNA. The distinction between the type of nucleotide is slightly more difficult: 74% of the proteins predicted to bind DNA and 72% of the proteins predicted to bind RNA were correct. Slightly over 53% of the residues binding DNA and/or RNA were correctly predicted. These levels of performance are at least 3-fold higher than random.

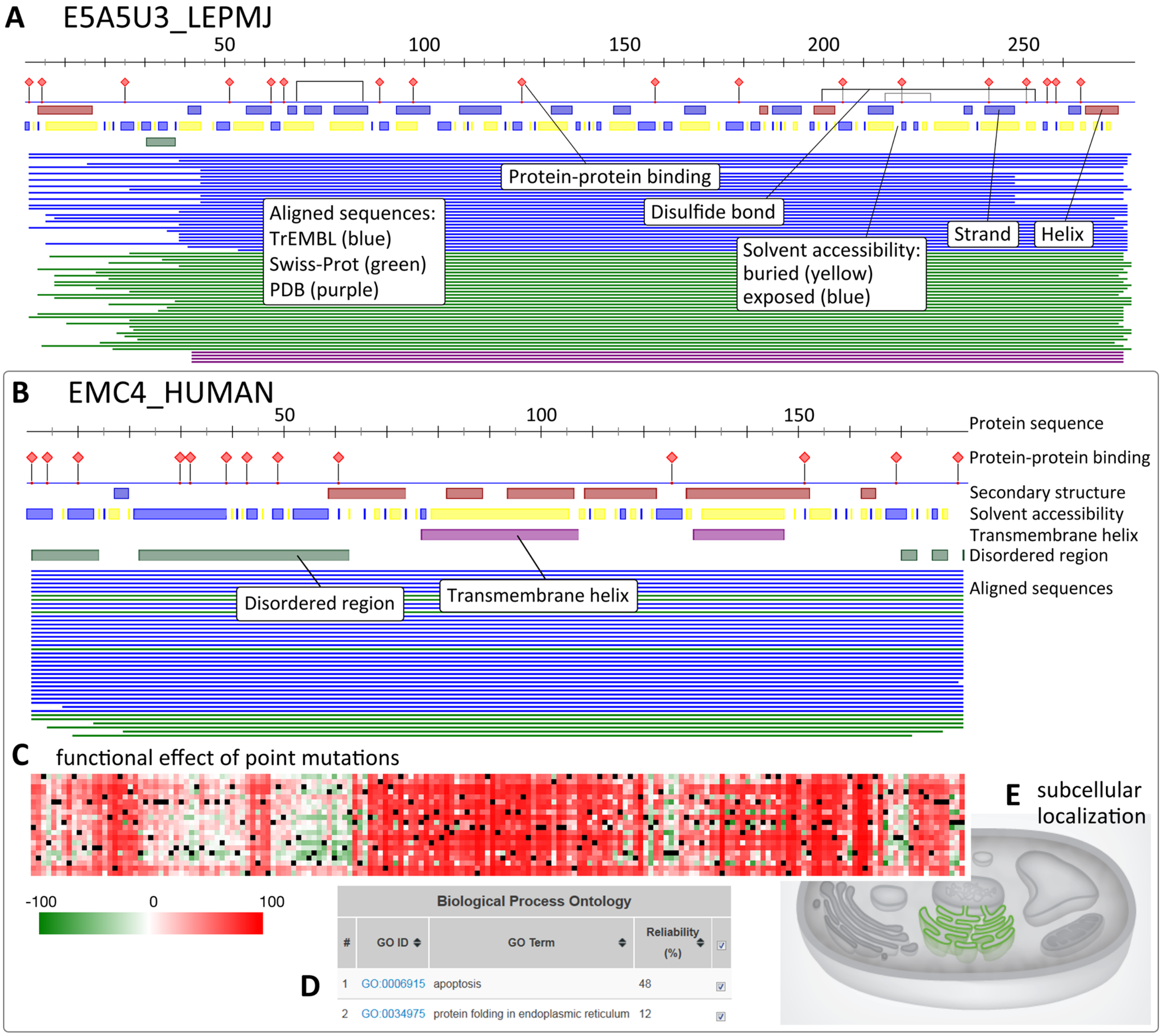
Recent: ConSurf conservation of surfaces explains function

ConSurf (26,27) estimates the evolutionary rate in protein families. These rates are useful for protein structure and function prediction because they reflect constrains imposed on the general evolutionary drift (10,15,55). Queried with a protein sequence, ConSurf first finds related sequences in UniProt (1). Evolutionary rates of amino acids are estimated based on evolutionary relatedness between the protein and its homologues using either empirical Bayesian (56) or maximum likelihood (57) methods. The strength of these methods is that they rely on the phylogeny of the sequences and thus can accurately distinguish between conservation due to short evolutionary time and conservation resulting from importance for maintaining protein foldability and function. If a structure is available, ConSurf maps the patterns of conservation upon the 3D structure. These patterns reveal crucial details about protein function.

WEB SERVER—UPDATES AND SOFTWARE

Graphical front-end

The dashboard page of PredictProtein results uses the BioJS (58) FeatureViewer component to show protein features (Figure 1A and B). Along the protein sequence, features



		Biological Process Ontology		
#	GO ID	GO Term	Reliability (%)	☑
1	GO:0006915	apoptosis	48	☑
2	GO:0034975	protein folding in endoplasmic reticulum	12	☑

Figure 1. Visual results from PredictProtein (PP). The PP Dashboard Viewer shows a schematic of all position-based predictions and sequence alignments. (**A**) Putative protein (UniProt AC E5A5U3). (**B**) ER membrane protein complex subunit 4 (EMC4, UniProt AC Q5J8M3). The protein sequence is represented by a scale on top of the predicted features. Features presented include protein–protein binding sites (ISIS2), disulfide bonds (DISULFIND), structural features such as secondary structure state and solvent accessibility (PROFphd), transmembrane helices (TMSEG) and disordered regions (MD). Proteins aligned by PSI-BLAST (7) are shown as thin lines colored by database origin (PDB (11), Swiss-Prot (12) and TrEMBL (1)). Clicking on each line links to the database entry of the hit. For all elements, tooltips disclose the annotated feature, its position in the sequence and its type (prediction versus database search). (**C**) A complete analysis of the functional effect of point mutations on EMC4 shown in a heatmap (SNAP2). (**D**) Predicted GO terms (metastudent) for EMC4 in tabular format. (**E**) The predicted cellular compartment, ER membrane, for EMC4 (LocTree3) is highlighted in green in a schematic of a eukaryotic cell.

are indicated by color and single residue pins. Depending on the protein, the overview features may include predictions of secondary structure and solvent accessibility, transmembrane helices, disulfide bonds and disordered regions. Details are available by zooming-in on local regions. Other views present additional annotations and predictions, e.g. functional landscapes of the effect of point mutations (SNAP2, Figure 1C), predicted GO terms (metastudent, Figure 1D) or subcellular localization (LocTree3, Figure 1E). In the dashboard viewer, users can mouse over the different view landmarks to reveal more information on the annotations.

The website features a Help section that includes interactive and instructive presentations. Each result section also provides a Help tab with specific explanations. All result pages feature an interactive Export menu for the download of selected raw data, as well as of the compiled archive with all data generated by the server. Additionally, we provide machine-readable output in XML and JSON. Output formatted for web presentations is available (HTML link at top right corner of main result page). The HTML view—most familiar to long-time users—aggregates results from most of the integrated methods in one page. This page also contains information that has not been integrated into the graphical view—yet—including results generated by some component methods and prediction confidence values. While we are working on the integration of all results into the graphical view, we highly encourage users to inspect this 'raw' HTML view. Finally, output is also available in text format (TEXT link, top right corner of results).

PPcache: pre-calculated results versus interactive jobs

One of the most beneficial recent resources from PredictProtein is the PPcache—a database that currently holds pre-calculated results for 11.7 million unique proteins—including all proteins of model organisms. If pre-calculated results are available for a PredictProtein query in PPcache, these are immediately returned. For results older than three months, users are given the option to re-run the query, thereby updating the PPcache. If no result exists in the PPcache, the job is processed, and users are notified upon job completion. PPcache currently requires roughly 100TB of disk space. We plan to open this repository for public access through a specialized API.

Downloadable software: packages and cloud-ready virtual machine

For full proteome analysis we make the full PredictProtein software suite available for download to be run either by installing the software packages on local machines or by deploying a virtual machine image in the cloud. Most methods from the PredictProtein pipeline are now available as open-source packages and are freely distributed through Debian (59) and Ubuntu. Following the Debian guidelines enforces best practices for software development and distribution and guarantees robustness, usability and maintainability of our software packages.

Users with access to cloud computing can download the PredictProtein Machine Image or PPMI (60), a disk image optimized for deployment in the cloud. The PPMI is bootable on server instances in cloud infrastructure services, or on locally installed virtualization software.

USE CASE

We demonstrate the usability and properties of PredictProtein through a simple example, the human endoplasmatic reticulum (ER) membrane protein complex subunit 4 (EMC4, UniProt AC Q5J8M3; Figure 1B–E). EMC4 is a small alpha-helical transmembrane protein with 183 residues. It is relatively well annotated, localizes to the membrane of the ER and is implicated in apoptosis (61,62).

The dashboard view of PredictProtein reveals an N-terminal disordered region of ∼60 residues (Figure 1B) interrupted by a short beta-strand (residues 17–20). This mainly disordered region is followed by a region dominated by alpha-helices. In this region, two transmembrane helices are predicted. Note that mouse-over can reveal annotations. The lines below the predictions sketch proteins with similar sequence. EMC4 is highly conserved, and nearly identical proteins are found in several mammalian organisms. Interestingly, the heatmap of functional effects (SNAP2) shows that the beta-strand interrupting the N-terminal disordered region and the transmembrane helices are highly sensitive to point mutations (Figure 1C). LocTree3 and metastudent predictions, respectively, agree at high reliability with the experimental subcellular localization of EMC4 in the ER membrane and its function in apoptosis (61,62) (Figure 1D and E). Additionally, metastudent identifies 'protein folding in endoplasmic reticulum' as biological function (Figure 1D; directed graph of predicted GO terms in Supplementary Figure S1, Supporting Online Material). This has already been shown for the yeast EMC4 (63).

The EMC4 example shows how users could have suspected some of those findings that have been experimentally verified (transmembrane helices, apoptosis, ER localization). On the other hand, it also suggests additional insights that might trigger new experiments, e.g. the importance of the disordered N-terminus, and the importance of the beta-strand that breaks it. May be this will provide more detail on the suggested involvement in protein folding and in apoptosis (Figure 1D (62)).

CONCLUSION

Over its 22 year existence, the PredictProtein server has substantially expanded. What started as a service to annotate some aspects of protein structure (secondary structure, solvent accessibility and transmembrane helices) has evolved into a comprehensive suite of methods important for the prediction of protein structural and functional features. It provides a single-point access to many original important results. Our focus on making reliable methods available and our technical focus on keeping our server useful to the community have sustained many challenges in an environment of low funding, growing use and increasing data deluge. Yet we continue finding ways to present our results efficiently and without overloading users from a wide variety of backgrounds and needs. The results pages aspire to give visually intuitive, unified presentations for most of the structural and functional annotations. The PredictProtein web server can help when little is known about the protein in question. For medium-to-high throughput analyses, users will find the publicly available, downloadable software packages and the PPMI a suitable option. For approximately every second query, our PPcache repository provides results immediately.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGMENTS

We acknowledge all who have contributed ideas, methods and components, as well as those who tested and documented bugs and provided insight and advice. So many of you users out there: thanks! Please see the full list of contributors in Table S2, Supporting Online Material and on our website http://ppopen.rostlab.org/credits. Thanks also to the following ROSTLAB members for their help: Tim

Karl for system maintenance, Milot Mirdita for helpful discussions, Marlena Drabik for handling administrative issues. Last, not least, thanks to all users who have been citing the usage of the service.

FUNDING

Alexander von Humboldt foundation through the German Ministry for Research and Education (BMBF: Bundesministerium fuer Bildung und Forschung).
Conflict of interest statement. None declared.

REFERENCES

1. Magrane,M. and Consortium,U. (2011) UniProt Knowledgebase: a hub of integrated protein data. *Database (Oxford)*, **2011**, bar009.
2. Bromberg,Y., Yachdav,G., Ofran,Y., Schneider,R. and Rost,B. (2009) New in protein structure and function annotation: hotspots, single nucleotide polymorphisms and the 'Deep Web'. *Curr. Opin. Drug Discov. Devel.*, **12**, 408–419.
3. Joosten,R.P., te Beek,T.A., Krieger,E., Hekkelman,M.L., Hooft,R.W., Schneider,R., Sander,C. and Vriend,G. (2011) A series of PDB related databases for everyday needs. *Nucleic Acids Res.* , **39**, D411–D419.
4. Moult,J., Fidelis,K., Kryshtafovych,A., Rost,B. and Tramontano,A. (2009) Critical assessment of methods of protein structure prediction-Round VIII. *Proteins*, **77**, 1–4.
5. Rost,B. and Sander,C. (1995) Progress of 1D protein structure prediction at last. *Proteins: Struct. Funct. Genet.*, **23**, 295–300.
6. Rost,B. and Sander,C. (1993) Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.*, **232**, 584–599.
7. Altschul,S.F., Madden,T.L., Schaffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic acids Res.*, **25**, 3389–3402.
8. Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,DJ (1990) Basic local alignment search tool. *J Mol Biol.*, **215**, 403–410.
9. Jones,D.T. (1999) Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.*, **292**, 195–202.
10. Przybylski,D. and Rost,B. (2002) Alignments grow, secondary structure prediction improves. *Proteins*, **46**, 197–205.
11. Berman,H.M., Westbrook,J.., Feng,Z., Gilliland,G., Bhat,T.N., Weissig,H., Shindyalov,I.N. and Bourne,PE(2000) The Protein Data Bank. *Nucleic Acids Res.*, **28**, 235–242.
12. Bairoch,A., Boeckmann,B., Ferro,S. and Gasteiger,E. (2004) Swiss-Prot: juggling between evolution and stability. *Brief. Bioinform.*, **5**, 39–55.
13. Sigrist,C.J., de Castro,E., Cerutti,L., Cuche,B.A., Hulo,N., Bridge,A., Bougueleret,L. and Xenarios,I. (2013) New and continuing developments at PROSITE. *Nucleic Acids Res.*, **41**, D344–D347.
14. Punta,M., Coggill,P.C., Eberhardt,R.Y., Mistry,J., Tate,J., Boursnell,C., Pang,N., Forslund,K., Ceric,G., Clements,J. *et al.* (2012) The Pfam protein families database. *Nucleic Acids Res.*, **40**, D290–D301.
15. Rost,B. (1996) PHD: predicting one-dimensional protein structure by profile-based neural networks. *Methods Enzymol.*, **266**, 525–539.
16. Rost,B. (2001) Review: protein secondary structure prediction continues to rise. *J. Struct. Biol.*, **134**, 204–218.
17. Bigelow,H. and Rost,B. (2006) PROFtmb: a web server for predicting bacterial transmembrane beta barrel proteins. *Nucleic Acids Res.*, **34**, W186–W188.
18. Lupas,A., Van Dyke,M. and Stock,J. (1991) Predicting coiled coils from protein sequences. *Science*, **252**, 1162–1164.
19. Ceroni,A., Passerini,A., Vullo,A. and Frasconi,P. (2006) DISULFIND: a disulfide bonding state and cysteine connectivity prediction server. *Nucleic Acids Res.*, **34**, W177–W181.
20. Wootton,J.C. and Federhen,S. (1996) Analysis of compositionally biased regions in sequence databases. *Methods Enzymol.*, **266**, 554–571.
21. Schlessinger,A., Punta,M. and Rost,B. (2007) Natively unstructured regions in proteins identified from contact predictions. *Bioinformatics*, **23**, 2376–2384.
22. Schlessinger,A., Liu,J. and Rost,B. (2007) Natively unstructured loops differ from other loops. *PLoS Comput. Biol.*, **3**, e140.
23. Schlessinger,A. and Rost,B. (2005) Protein flexibility and rigidity predicted from sequence. *Proteins*, **61**, 115–126.
24. Schlessinger,A., Yachdav,G. and Rost,B. (2006) PROFbval: predict flexible and rigid residues in proteins. *Bioinformatics*, **22**, 891–893.
25. Schlessinger,A., Punta,M., Yachdav,G., Kajan,L. and Rost,B. (2009) Improved disorder prediction by combination of orthogonal approaches. *PLoS One*, **4**, e4433.
26. Ashkenazy,H., Erez,E., Martz,E., Pupko,T. and Ben-Tal,N. (2010) ConSurf 2010: calculating evolutionary conservation in sequence and structure of proteins and nucleic acids. *Nucleic Acids Res.*, **38**, W529–W533.
27. Celniker,G., Nimrod,G., Ashkenazy,H., Glaser,F., Martz,E., Mayrose,I., Pupko,T. and Ben-Tal,N. (2013) ConSurf: using evolutionary data to raise testable hypotheses about protein function. *Israel J. Chem.*, **53**, 199–206.
28. Hecht,M., Bromberg,Y. and Rost,B. (2013) News from the protein mutability landscape. *J. Mol. Biol.*, **425**, 3937–3948.
29. Hamp,T., Kassner,R., Seemayer,S., Vicedo,E., Schaefer,C., Achten,D., Auer,F., Boehm,A., Braun,T., Hecht,M. *et al.* (2013) Homology-based inference sets the bar high for protein function prediction. *BMC Bioinformatics*, **14**(Suppl. 3), S7. doi:10.1186/1471-2105-14-S3-S7
30. Goldberg,T., Hecht,M., Hamp,T., Karl,T., Yachdav,G., Ahmed,N., Altermann,U., Angerer,P., Ansorge,S., Balasz,K. *et al.* (2014) LocTree3 prediction of localization. *Nucleic Acids Res.*, doi: 10.1093/nar/gku396.
31. Rost,B., Yachdav,G. and Liu,J. (2004) The PredictProtein server. *Nucleic Acids Res.*, **32**, W321–W326.
32. Tusnady,G.E., Dosztanyi,Z. and Simon,I. (2005) PDB_TM: selection and membrane localization of transmembrane proteins in the protein data bank. *Nucleic Acids Res.*, **33**, D275–D278.
33. Lomize,M.A., Lomize,A.L., Pogozheva,I.D. and Mosberg,H.I. (2006) OPM: orientations of proteins in membranes database. *Bioinformatics*, **22**, 623–625.
34. Petersen,T.N., Brunak,S., von Heijne,G. and Nielsen,H. (2011) SignalP 4.0: discriminating signal peptides from transmembrane regions. *Nat. Methods*, **8**, 785–786.
35. Kall,L., Krogh,A. and Sonnhammer,E.L. (2005) An HMM posterior decoder for sequence feature prediction that includes homology information. *Bioinformatics*, **21**(Suppl. 1), i251–i257.
36. Jones,D.T. (2007) Improving the accuracy of transmembrane protein topology prediction using evolutionary information. *Bioinformatics*, **23**, 538–544.
37. Hecht,M. (2011) *Technische Universität Muenchen (TUM)*, Munich, Germany.
38. Bromberg,Y. and Rost,B. (2007) SNAP: predict effect of non-synonymous polymorphisms on function. *Nucleic Acids Res.*, **35**, 3823–3835.
39. Bromberg,Y., Overton,J., Vaisse,C., Leibel,R.L. and Rost,B. (2009) In silico mutagenesis: a case study of the melanocortin 4 receptor. *Faseb J.*, **23**, 3059–3069.
40. Yachdav,G., Hecht,M., Yeheskel,A., Pasmanik-Chor,M. and Rost,B. (2014) HeatMapViewer:interactive display of 2D data in biology. *F1000Research*, **3**, doi:10.12688/f1000research.3-48.v1.
41. Goldberg,T., Hamp,T. and Rost,B. (2012) LocTree2 predicts localization for all domains of life. *Bioinformatics*, **28**, i458–i465.
42. Ashburner,M., Ball,C.A., Blake,J.A., Botstein,D., Butler,H., Cherry,J.M., Davis,A.P., Dolinski,K., Dwight,S.S., Eppig,J.T. *et al.* (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, **25**, 25–29.
43. Radivojac,P., Clark,W.T., Oron,T.R., Schnoes,A.M., Wittkop,T., Sokolov,A., Graim,K., Funk,C., Verspoor,K., Ben-Hur,A. *et al.* (2013) A large-scale evaluation of computational protein function prediction. *Nat. Methods*, **10**, 221–227.
44. Minneci,F., Piovesan,D., Cozzetto,D. and Jones,D.T. (2013) FFPred 2.0: improved homology-independent prediction of gene ontology terms for eukaryotic protein sequences. *PLoS One*, **8**, e63754.
45. Punta,M. and Rost,B. (2005) PROFcon: novel prediction of long-range contacts. *Bioinformatics*, **21**, 2960–2968.
46. Liu,J., Tan,H. and Rost,B. (2002) Loopy proteins appear conserved in evolution. *J. Mol. Biol.*, **322**, 53–64.

47. Liu,J. and Rost,B. (2003) NORSp: predictions of long regions without regular secondary structure. *Nucleic Acids Res.*, **31**, 3833–3835.
48. Dosztanyi,Z., Csizmok,V., Tompa,P. and Simon,I. (2005) IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. *Bioinformatics*, **21**, 3433–3434.
49. Ward,J.J., McGuffin,L.J., Bryson,K., Buxton,B.F. and Jones,D.T. (2004) The DISOPRED server for the prediction of protein disorder. *Bioinformatics*, **20**, 2138–2139.
50. Mizianty,M.J., Stach,W., Chen,K., Kedarisetti,K.D., Disfani,F.M. and Kurgan,L. (2010) Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources. *Bioinformatics*, **26**, i489–i496.
51. Sickmeier,M., Hamilton,J.A., LeGall,T., Vacic,V., Cortese,M.S., Tantos,A., Szabo,B., Tompa,P., Chen,J., Uversky,V.N. *et al.* (2007) DisProt: the database of disordered proteins. *Nucleic Acids Res.*, **35**, D786–D793.
52. Ofran,Y. and Rost,B. (2007) ISIS: interaction sites identified from sequence. *Bioinformatics*, **23**, e13–e16.
53. Hamp,T. and Rost,B. (2012) Alternative protein-protein interfaces are frequent exceptions. *PLoS Comput. Biol.*, **8**, e1002623.
54. Hönigschmid,P. (2012) *Diploma thesis*, Technische Universität München,Munich, Germany.
55. Rost,B., Liu,J., Nair,R., Wrzeszczynski,K.O. and Ofran,Y. (2003) Automatic prediction of protein function. *Cell. Mol. Life Sci.*, **60**, 2637–2650.
56. Mayrose,I., Graur,D., Ben-Tal,N. and Pupko,T. (2004) Comparison of site-specific rate-inference methods for protein sequences: empirical Bayesian methods are superior. *Mol. Biol. Evol.*, **21**, 1781–1791.
57. Pupko,T., Bell,R.E., Mayrose,I., Glaser,F. and Ben-Tal,N. (2002) Rate4Site: an algorithmic tool for the identification of functional regions in proteins by surface mapping of evolutionary determinants within their homologues. *Bioinformatics*, **18**(Suppl. 1), S71–S77.
58. Gomez,J., Garcia,L.J., Salazar,G.A., Villaveces,J., Gore,S., Garcia,A., Martin,M.J., Launay,G., Alcantara,R., Del-Toro,N. *et al.* (2013) BioJS: an open source JavaScript framework for biological data visualization. *Bioinformatics*, **29**, 1103–1104.
59. Moller,S., Krabbenhoft,H.N., Tille,A., Paleino,D., Williams,A., Wolstencroft,K., Goble,C., Holland,R., Belhachemi,D. and Plessy,C. (2010) Community-driven computational biology with Debian Linux. *BMC Bioinformatics*, **11**(Suppl. 12), S5. doi:10.1186/1471-2105-11-S12-S5
60. Kajan,L., Yachdav,G., Vicedo,E., Steinegger,M., Mirdita,M., Angermuller,C., Bohm,A., Domke,S., Ertl,J., Mertes,C. *et al.* (2013) Cloud prediction of protein structure and function with PredictProtein for Debian. *Biomed. Res. Int.*, **2013**, 398968. doi: 10.1155/2013/398968
61. Christianson,J.C., Olzmann,J.A., Shaler,T.A., Sowa,M.E., Bennett,E.J., Richter,C.M., Tyler,R.E., Greenblatt,E.J., Harper,J.W. and Kopito,R.R. (2012) Defining human ERAD networks through an integrative mapping strategy. *Nat. Cell Biol.*, **14**, 93–105.
62. Ring,G., Khoury,C.M., Solar,A.J., Yang,Z., Mandato,C.A. and Greenwood,M.T. (2008) Transmembrane protein 85 from both human (TMEM85) and yeast (YGL231c) inhibit hydrogen peroxide mediated cell death in yeast. *FEBS Lett.*, **582**, 2637–2642.
63. Jonikas,M.C., Collins,S.R., Denic,V., Oh,E., Quan,E.M., Schmid,V., Weibezahn,J., Schwappach,B., Walter,P., Weissman,J.S. *et al.* (2009) Comprehensive characterization of genes required for protein folding in the endoplasmic reticulum. *Science*, **323**, 1693–1697.

Supporting online material for: PredictProtein – open online prediction of protein structure and function

Guy Yachdav, Edda Kloppmann, Laszlo Kajan, Max Hecht, Tatyana Goldberg, Tobias Hamp, Andrea Schafferhans, Burkhard Rost et al

Table S1: Methods incorporated into PredictProtein

Category	Method/dB Name	Description	Command line tool		Output available in		
			Name	Version	Visual	HTML	TEXT
Database search	BLAST	Pairwise alignment	blastall	2.2.26	-	-	x
Database search	PSI-BLAST	Profile based alignment	blastpgp	2.2.27	x	x	x
Database search	HMMER	Hidden markov model search	hmm2pfam	2.3.2	-	-	x
Database search	ps_scan	PROSITE scanning program	ps_scan.pl	1.67	-	x	x
Database Search	PSSH	Mapping of sequence positions onto a structure	generate_pssh2	1.0.0	-	-	x
Database Search	Species Mapper	Maps organism code to kingdom	speciesmapper	1.0,1	x	-	-
Database Search	ID Mapper	maps ids across major databases	idmapper	1.0.3	x	-	-
Analysis method	SEG	Low complexity regions markup	ncbi-seg	0.0.20000620	-	x	x
Analysis method	NCOILS	Calculates the probability that the sequence will adopt a coiled-coil conformation	ncoils	2002	-	x	x
Analysis method	HSSP	Homology derived secondary Structure of proteins	hssp_filter	1	-	-	x
Prediction method	NORSp	NOn-Regular Secondary Structure	norsp	1.0.3	-	x	x
PHDhtm	PHDhtm	Prediction of membrane helices	phd.pl	1.0.40	x	x	x
Prediction method	TMSEG	Prediction of membrane helices	tmseg	1.0.0	x	x	x
Prediction method	PROFsec	Prediction of secondary structure state	prof	1.0.40	x	x	x
Prediction method	PROFacc	Prediction of solvent accessbility	prof	1.0.40	x	x	x
Prediction Method	Reprof	Improved prediction of secondary structure state	reprof	1.0.0	-	-	x
Prediction method	PROFtmb	Prediction of transmembrane beta-barrels	proftmb	1.1.12	x	x	x
Prediction method	DISULFIND	Prediction of disulfide bridges	disulfinder	1.2	x	x	x
Prediction method	PROFBval	Prediction of residue mobility	profbval	1.0.16	x	x	x

Category	Method/dB Name	Description	Command line tool		Output available in		
			Name	Version	Visual	HTML	TEXT
Prediction method	NorsNet	Prediction protein disordered sites	norsnet	1.0.16	x	x	x
Prediction method	UCON	Contact based prediction of disordered sites	ucon	1.0.8	x	x	x
Prediction method	Meta-Disorder	Consensus based prediction of protein disorder	metadisorder	1.0.14	x	x	x
Prediction method	ISIS2	Prediction of protein-protein interaction sites	profisis2	1.0.0	x	x	x
Prediction method	SomeNA	Prediction of protein –DNA, -RNA binding sites	somena	1.0.0	X	-	-
Prediction method	LocTree3	Prediction of sub-cellular localization for all domains of life	loctree3	1.0.5	x	x	x
Prediction method	PredictNLS	Prediction of Nuclear Localization Signals (NLS)	predictnls	1.0.18	-	x	x
Prediction method	metastudent	Prediction of GO terms for Molecular Function and Biological Process	metastudent	1.0.9	x	-	X
Prediction method	SNAP2	Prediction of functional changes due to single nucleotide polymorphism	snap2	1.0.10	x	-	x
Prediction method	ConSurf	Identification of functional sites	consurf	1.0.0	x	-	x
Database	UniRef	Clustered set of sequences	N/A	Updates monthly	x	x	X
Database	BIG	non-redundant combination of Swiss-Prot, TrEMBL, PDB	N/A	Updates monthly	x	x	x
Database	PDB	Repository of protein structures	N/A	Updates monthly	x	x	x
Database	Pfam-A	Protein families	N/A	Updates quarterly	x	x	x
Database	PROSITE	Database of biologically significant sites, patterns and profiles	N/A	Updates quarterly	x	x	X

Table S2: List of contributors

This table lists all non-coauthors contributors. All contributors are acknowledged at http://ppopen.rostlab.org/credits

Name	Contribution	Affiliation
Jinfeng Liu	Contributed code for the PredictProtein pipeline Contributed the NORS, CHOP & CHOPnet (discontinued) methods	Alumnus
Yanay Ofran	Contributed the ISIS and DISIS methods (discontinued)	Alumnus
Rajesh Nair	Contributed the LocTree method (discontinued)	Alumnus
Henry Bigelow	Contributed the PROFtmb method	Alumnus
Sven Mika	Provided the UniqueProt method	Alumnus
Dariusz Przybylski	Contributed the AGAPE method (discontinued)	Alumnus
Kazimierz Wrzeszczynski	Contributed code and ideas	Alumnus
Paolo Frasconi	Contributed the DISULFIND method	External contributor
Antoine de Daruvar	Helped getting the first PredictProtein server online	Original contributor
Roy Omond	Helped in the communication between VMS and Unix systems for the first server	Original contributor
Jonas Reeb	Member of the Scientific Editorial Board Responsible for transmembrane annotations	Scientific board
Juan Miguel Cejuela	Contributed the literature search component	Scientific board
Rachel First	Designed the artwork for the localization prediction Designed the site tutorial	Graphics design
Thomas Splettstoesser	Designed the PredictProtein logo	Graphics design

Figure S1

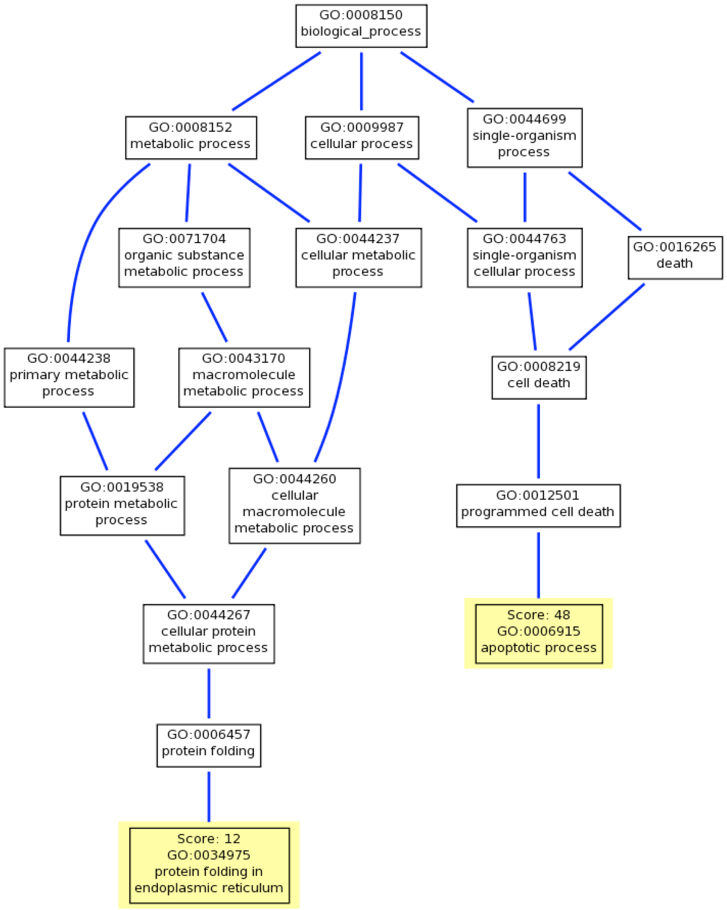


Figure S1: GO term predictions. GO term predictions from metastudent are presented in PredictProtein in tabular form (Fig. 1D) and as directed graph as shown here through the example of human EMC4 (UniProt AC Q5J8M3).

WEB TOOL

HeatMapViewer: interactive display of 2D data in biology [v1; ref status: indexed, http://f1000r.es/2u6]

Guy Yachdav[1-3], Maximilian Hecht[1,2], Metsada Pasmanik-Chor[4], Adva Yeheskel[4], Burkhard Rost[1-3]

[1]TUM, Department of Informatics, Bioinformatics & Computational Biology, 5748 Garching/ Munich, Germany
[2]TUM Graduate School of Information Science in Health (GSISH), 85748 Garching/Munich, Germany
[3]Biosof LLC, New York, NY, 10001, USA
[4]Bioinformatics Unit, G.S.W. Faculty of Life Sciences, Tel Aviv University, Tel Aviv, 69978, Israel

v1 **First published:** 13 Feb 2014, **3**:48 (doi: 10.12688/f1000research.3-48.v1)
Latest published: 13 Feb 2014, **3**:48 (doi: 10.12688/f1000research.3-48.v1)

Abstract

Summary: The HeatMapViewer is a BioJS component that lays-out and renders two-dimensional (2D) plots or heat maps that are ideally suited to visualize matrix formatted data in biology such as for the display of microarray experiments or the outcome of mutational studies and the study of SNP-like sequence variants. It can be easily integrated into documents and provides a powerful, interactive way to visualize heat maps in web applications. The software uses a scalable graphics technology that adapts the visualization component to any required resolution, a useful feature for a presentation with many different data-points. The component can be applied to present various biological data types. Here, we present two such cases – showing gene expression data and visualizing mutability landscape analysis.

Availability: https://github.com/biojs/biojs; http://dx.doi.org/10.5281/zenodo.7706.

BioJS This article is included in the BioJS

Open Peer Review

Referee Status: ✓ ✓

	Invited Referees	
	1	**2**
version 1 published 13 Feb 2014	✓ report	✓ report

1 **Jordi Deu-Pons**, University Pompeu Fabra (UPF) Spain

2 **John Ison**, EMBL European Bioinformatics Institute UK

Discuss this article

Comments (0)

Corresponding author: Guy Yachdav (gyachdav@rostlab.org)

How to cite this article: Yachdav G, Hecht M, Pasmanik-Chor M *et al.* ***HeatMapViewer*: interactive display of 2D data in biology [v1; ref status: indexed, http://f1000r.es/2u6]** *F1000Research* 2014, **3**:48 (doi: 10.12688/f1000research.3-48.v1)

Grant information: GY and MH are supported by a grant from the Alexander von Humboldt foundation through the German Ministry for Research and Education (BMBF: Bundesministerium fuer Bildung und Forschung).
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: No competing interests were disclosed.

First published: 13 Feb 2014, **3**:48 (doi: 10.12688/f1000research.3-48.v1)
First indexed: 05 Mar 2014, **3**:48 (doi: 10.12688/f1000research.3-48.v1)

Introduction

Biological data are often organized into matrices in which the rows signify different items of interest (a gene, a subject, a probe or a position in a sequence), while the columns describe different experiments, variations, or samples. Matrices are easy to process by algorithms. In contrast, the details in large matrices are often, at best, challenging for experts who want to "understand" the data. The information in matrices is usually better digested if presented by 3D plots or heat maps. Heat maps are essentially simplified versions of 3D plots that replace the 3rd dimension with color gradients, thereby conveniently displaying the information contained in matrices. Such heat maps allow for easy visual differentiation between high and low values in a matrix.

Such heat maps are, for example, commonly used to display microarray data as they quickly show which genes (rows) are differentially expressed under some conditions (columns). Microarray technologies utilize arrays of probes located on different exons for each gene and can be helpful in determining gene function by measuring transcription and translation levels under certain experimental conditions. The expression values for the differential expression may be presented at the exon level, correlated with protein domains, and may help to decipher a complex gene expression pattern.

Heat maps can also display the effect of point mutations (single amino acid substitutions, or non-synonymous Single Nucleotide Polymorphisms – nsSNPs). Through the application of methods that predict the impact of mutations[1–4] we can expand from the view of single variants to the level of sketching the entire *mutability landscape*[5]. This *mutability landscape* is defined by the impact of substituting every residue at each position in a protein by each of the 19 non-native amino acids. The resulting predictions can then be shown in a heat map in order to visualize the impact of each substitution. Regions where mutations have a high average effect (i.e. where almost every substitution is predicted to alter protein function) are especially interesting as these are likely to be of particular and direct importance for protein function.

We developed *HeatMapViewer* as a BioJS component that can easily be used, reused and, if needed, extended to display matrix data. BioJS[6] is an open source JavaScript library of components for visualization of biological data on the web. As a JavaScript component, the *HeatMapViewer* is very flexible, interactive and webready. Previous libraries generating graphical HeatMaps render either static images[7] or are highly specialized and cannot be reused[8]. To the best of our knowledge, this is the first client-side modular component to visualize matrices that can be integrated into other web applications in a standard manner.

The *HeatMapViewer* component

HeatMapViewer uses the D3[9] JavaScript library to render Scalable Vector Graphics (SVG) objects. SVG technology is now standardized and native to modern web browsers (e.g. Mozilla, Chrome, Safari). The component accepts a simple JSON object containing the data matrix that will be rendered. A secondary JSON object contains configuration directions such as the target DIV element onto which the component will be rendered, the data range to be shown, the color scheme to be used for the component, the size of the canvas showing the component and the minimum cell size (by default these last two options can be computed automatically).

The *HeatMapViewer* component automatically renders a heat map based on the input data object and the pre-set color-scheme. Positioning and layout are automatically calculated given the available browser window size. If presenting the entire heat map requires individual cells to be smaller than a given threshold, a secondary panel is automatically rendered to show a zoomed-in version of a local segment in the heat map. This zoom-in panel is presented right under the main heat map panel. The labels for the X-axis and Y-axis are laid out above the top row and next to the left column. The component provides a user control in the form of a frame that can be dragged along the main heat map to determine which area of the heat map should appear in the zoom-in panel. Additionally, a scale bar is presented to show the value ranges and which colors correspond to those values. Finally, each cell in the heat map is associated with a mouse-over event that pops-up tooltips showing the data-value of the cell.

The *HeatMapViewer* component can be obtained from the BioJS registry at https://github.com/biojs/biojs. For users wishing to test the component's capabilities to generate heat map plots for their data without downloading and installing the component, we have set up a server: http://www.rostlab.org/services/heatmap-viewer. The server allows users to upload their data in Comma Separated Values (CSV) format and then renders a heat map on the screen. The server also allows exporting the resulting graphics rendering it into an image.

Application use-cases and examples

Eye disease Retinitis Pigmentosa (RP)

The rhodopsin gene encodes a protein of the outer photoreceptor segment that is essential for the visual transduction cascade. Since 1989, many mutations in the rhodopsin gene have been found to be involved in the eye disease Retinitis Pigmentosa (also known as Retinopathia pigmentosa or simply RP[10]). RP is a hereditary disease causing retinal degeneration and thereby destroying photoreceptors; this results in severe vision impairment or even blindness.

A typical study of such a hereditary disease might begin with a protocol as follows. According to the UCSC genome browser[11], human rhodopsin (RHO, RefSeq: NM_000539.3) consists of 5 exons (located on chr3:129,247,482-129,254,187). The total gene length is 6706 bps (base pairs/nucleotides). The coding region (chr3:129,247,577-129,252,561; i.e. extending over 4985 bps), is translated into a gene-product/protein with 348 residues (UniProt identifier: P08100[12], SwissProt identifier: OPSD_HUMAN[13]). This protein has a single large domain (Pfam identifier: PF00001[14]) that is dominated by a "standard" 7-transmembrane receptor region (rhodopsin family), which spans most of the coding region (residues 55 - 306). The human rhodopsin is highly expressed in the heart, liver, skeletal muscle, thyroid and the eye retina.

Viewing gene expression data

It is interesting to locate the array probes intensities on the various protein domain regions. We map the expression profiles of the RHO (from GEO43134) to the structural protein regions through visualization with the *HeatMapViewer* component (Figure 1). The different experimental conditions are presented on the rows, while the probes for the RHO gene are shown on the columns, annotated with exon and trans-membrane (TM) location. Probes with high expression are marked in red; those with low expression are colored green. The differences in color of the same probe along the different conditions provides useful information concerning the expression intensity of the various probes, and possible variations in alternative splicing patterns and region conservation across the different samples.

Predicted protein mutability landscape

Since RP is caused by mutations in the rhodopsin gene, researchers have extensively investigated different variations of the gene. Thus, up to now over 100 mutations have been identified and associated with RP. More generally: single nucleotide variations constitute most of the genetic variation among humans and therefore play an important role when studying hereditary diseases or differential drug response. In this context, we show another possible application of the *HeatMapViewer*, again using the 7TM human rhodopsin (SwissProt identifier: OPSD_HUMAN[13]). The *HeatMapViewer* provides a fast and easy way to represent high dimensional data in a visually comprehensible way that immediately conveys where mutations are likely to be deleterious. Without using a tool such as the *HeatMapViewer*, we could hardly obtain an overview of the protein mutability landscape[5]. Mutability landscape studies involve predicting the effect of all possible nsSNPs through computational methods, visualizing the predictions in heat maps and cross-linking these predictions with additional sources of information (such as secondary structure, active sites and correlated mutational behavior). Such regions might highlight important aspects of RP. To this end, heat maps (Figure 2b, 2c) can easily distinguish between low effect regions (represented in blue) and high effect regions (represented in red) while additional information (such as the secondary structure; Figure 2a) can simply be over-laid. These two components already perfectly convey the information that high effect regions are mainly found in the transmembrane helices and in close proximity of the binding sites. Displaying this simple fact without a heat map would be daunting due to the high dimensionality of the underlying data.

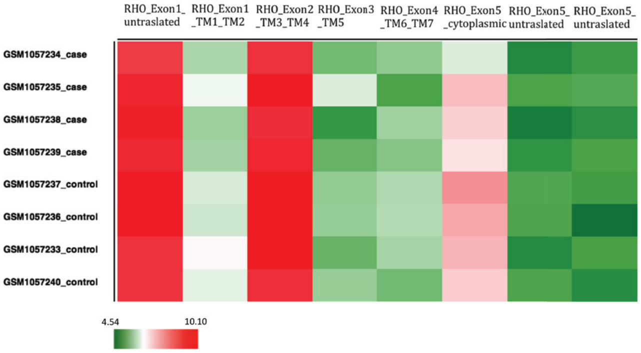


Figure 1. *HeatMapViewer* component visualization of microarray expression experiment (Korir *et al.* 2012; GSE43134). In this experiment, a mutation in a splicing factor that causes Retinitis Pigmentosa (RP) was shown to have an effect on mRNA splicing. Moreover, mutations in the rod photoreceptor-specific protein rhodopsin (RHO) are known to cause RP. Log2 expression values for the 8 probes of human RHO were obtained and located to each of its 5 exons and the 7 trans-membrane (TM) regions (columns). It is interesting to note that the different probes (located on the various regions of RHO), are differentially expressed (high expression colored red and low expression in green). Moreover, we can observe that some RHO probes are expressed differently in the control than in the treatment (case, rows). These results may indicate the effect of the mutated splicing factor on RHO gene in RP disease.

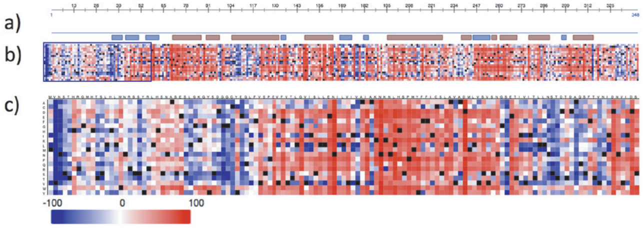


Figure 2. The *HeatMapViewer* component displays the mutability landscape of OPSD_HUMAN. Panel **a**) sketches the secondary structure (helices in red, beta strands in blue). Panel **b**) shows the predictions of effects for each amino acid substitution. Effects are depicted as color intensities ranging from dark blue (high probability of no or little effect) over white (effect can not be predicted or only with very low reliability) to dark red (high probability of strong effects). Black depicts wildtype residues. The blue box marks the zoomed-in region shown in panel **c**).

Conclusions

The *HeatMapViewer* component provides a new, powerful way to generate and display matrix data in web presentations and in publications. The use of scalable graphics enables the rendering of high-resolution images as the interactive nature of the component permits those graphics to be scaled on-demand. Furthermore the component can be applied to different cases highlighting various points of interest from gene expression levels to the effects of mutability on protein function. Finally, to make the *HeatMapViewer* component widely accessible, we set up a public web server to which users can upload their matrix data and use the resulting code to show an interactive heat map.

Software availability

Zenodo: HeatMap Viewer, doi: 10.5281/zenodo.7706[15]

GitHub: BioJS, https://github.com/biojs/biojs

Author contributions

GY developed the original *HeatMapViewer* component and maintains the *HeatMapViewer* service. MH contributed the SNP analysis use case. MPC and AY contributed the gene expression data use case. BR contributed the initial idea for the component and helped with its design.

Competing interests

No competing interests were disclosed.

Grant information

GY and MH are supported by a grant from the Alexander von Humboldt foundation through the German Ministry for Research and Education (BMBF: Bundesministerium fuer Bildung und Forschung).

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgements

Thanks to Tim Karl and Laszlo Kajan (TUM) for invaluable help with hardware and software; to Marlena Drabik (TUM) for administrative support; to thanks Edda Kloppman and Tatyana Goldberg (TUM) for helpful discussions and invaluable feedback.

References

1. Bromberg Y, Rost B: **SNAP: predict effect of non-synonymous polymorphisms on function.** *Nucleic Acids Res.* 2007; **35**(11): 3823–35.
PubMed Abstract | Publisher Full Text | Free Full Text
2. Bromberg Y, Overton J, Vaisse C, *et al.*: **In silico mutagenesis: a case study of the melanocortin 4 receptor.** *FASEB J.* 2009; **23**(9): 3059–69.
PubMed Abstract | Publisher Full Text | Free Full Text
3. Sunyaev S, Ramensky V, Koch I, *et al.*: **Prediction of deleterious human alleles.** *Hum Mol Genet.* 2001; **10**(6): 591–7.
PubMed Abstract | Publisher Full Text
4. Pauline CNg, Steven H: **SIFT: Predicting amino acid changes that affect protein function.** *Nucleic Acids Res.* 2003; **31**(13): 3812–4.
PubMed Abstract | Publisher Full Text | Free Full Text
5. Hecht M, Bromberg Y, Rost B: **News from the protein mutability landscape.** *J Mol Biol.* 2013; **425**(21): 3937–48.
PubMed Abstract | Publisher Full Text
6. Gómez J, García LJ, Salazar GA, *et al.*: **BioJS: an open source Javascript framework for biological data visualization.** *Bioinformatics.* 2013; **29**(8): 1103–4 2013.
PubMed Abstract | Publisher Full Text | Free Full Text
7. Pavlidis P, Noble WS: **Matrix2png: a utility for visualizing matrix data.** *Bioinformatics.* 2003; **19**(2): 295–6.
PubMed Abstract | Publisher Full Text
8. Xia J, Lyle NH, Mayer ML, *et al.*: **INVEX–a web-based tool for integrative visualization of expression data.** *Bioinformatics.* 2013; **29**(24): 3232–4.
PubMed Abstract | Publisher Full Text | Free Full Text
9. Heer J, Bostock M, Ogievetsky V: **D3: Data-Driven Documents.** *IEEE Trans Vis Comput Graph.* 2011; **17**(12): 2301–9.
PubMed Abstract | Publisher Full Text
10. Dryja TP, McGee TL, Reichel E, *et al.*: **A point mutation of the rhodopsin gene in one form of retinitis pigmentosa.** *Nature.* 1990; **343**(6256): 364–6.
PubMed Abstract | Publisher Full Text
11. Kent WJ, Sugnet CW, Furey TS, *et al.*: **The human genome browser at ucsc.** *Genome Res.* 2002; **12**(6): 996–1006.
PubMed Abstract | Publisher Full Text | Free Full Text
12. UniProt Consortium. **Update on activities at the universal protein resource (uniprot) in 2013.** *Nucleic Acids Res.* 2013; **41**(Database issue): D43–7.
PubMed Abstract | Publisher Full Text | Free Full Text
13. Boeckmann B, Blatter MC, Famiglietti L, *et al.*: **Protein variety and functional diversity: Swiss-Prot annotation in its biological context.** *C R Biol.* 2005; **328**(10–11): 882–99.
PubMed Abstract | Publisher Full Text
14. Punta M, Coggill PC, Eberhardt RY, *et al.*: **The Pfam protein families database.** *Nucleic Acids Res.* 2012; **40**(Database issue): D290–301.
PubMed Abstract | Publisher Full Text | Free Full Text
15. Yachdav G: **HeatMap Viewer.** *Zenodo.* 2014. Data Source