Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Subject Section

SSR-viz - a toolbox to detect and visualize protein subfamily specific residues

Paul Zierep [1,*], Stefan Günther [2]

[1] Pharmaceutical Bioinformatics, Institute of Pharmaceutical Science, Albert-Ludwigs-University, Hermann-Herder-Strasse 9, Freiburg 79104, Germany.

* To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Protein subfamilies share a common structure but differ in functionalities such as substrate specificity. These differences can often be assigned to a limited set of subfamily specific residues (SSRs). Often, only experimental evaluation, expert knowledge and detailed protein structure investigation, reveal the correct assignment of these SSRs. Here, we introduce the toolbox SSR-viz, which allows to detect and visualize these SSRs based on a multiple sequence alignment (MSA) and a file containing the subfamily class labels. The applied algorithm can be adjusted to a given problem and allows users to perform a thorough analysis of their sequences.

Results: Substrate specificity of non-ribosomal peptide synthases were analyzed. The detected residues are located in the binding side and are in accordance with previous performed studies.

Availability: The toolbox is written as a GUI in Python3 and is available via PyPi and the GitHub repository (https://github.com/PhaBiFreiburg/SSR-viz). Standalone executables are also available for Windows and Linux.

Contact: stefan.guenther@pharmazie.uni-freiburg.de

Supplementary information: Supplementary data are available at *Bioinformatics* online.

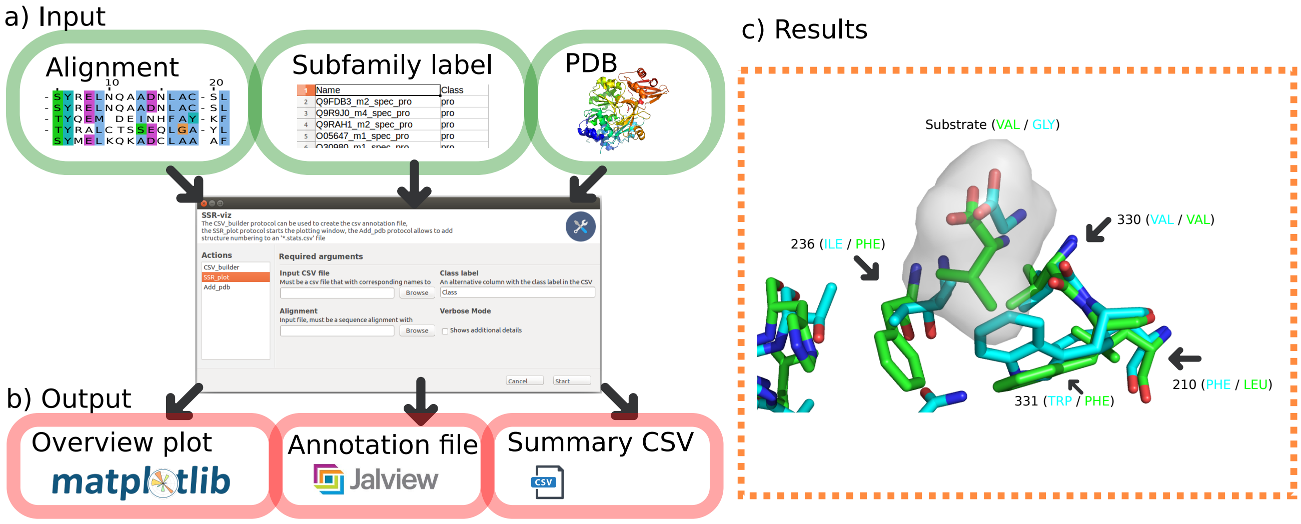
1 Introduction

Protein subfamilies differ from each other in very specific functions, such as substrate specificity, protein-protein interaction or reaction type. These functionalities can often be assigned to a limited set of residues (referred to as subfamily specific residues (SSRs) in this article). The identification of those SSRs is crucial to the understanding of protein subfamily diversity. SSRs with high statistical significance can form the basis for various experiments such as side directed mutagenesis and rational design of proteins.

Various approaches have been proposed to detect SSRs based on analysis of multiple sequence alignments (MSAs) of two or more protein subfamilies (Hannenhalli *et al.*, 2000;Edwards *et al.*, 2005; Olivera-Nappa *et al.*, 2011;Suplatov *et al.*, 2014;Suplatov *et al.*, 2015). All these approaches share the common objective: to identify positions in the MSA which are conserved in one protein subfamily but differ between subfamilies. Some algorithms also include physico-chemical properties of the residues in order to distinguish true SSRs from residues which are due to evolutionary mutations.

In order to allow researchers a thorough study of their proteins we implemented an algorithm which follows this common objective but is also flexible and can be adjusted to a specific protein family observed. The algorithm allows the choice of various substitution matrices to define the similarity of residues. The detection allows for three different kind of scoring modes: one-vs-one, which allows to detect SSRs most important to distinguish two subfamilies from each other; one-vs-all, which allows to detect SSRs most important to distinguish one subfamily from a set of subfamilies and all-vs-all, which detects the most important residues to distinguish various subfamilies from each other.

Additionally, SSR-viz provides assistance for various tasks commonly encountered on the search for SSRs. The main features are: The easy mapping of the alignment with a CSV-file which holds the information of the subfamily class labels. The addition of a window function, which allows to observe single amino acid residues as well as general areas of interest in an alignment. The residues can be mapped to a protein structure, which facilitates the further investigation in a protein visualization tool such as PyMol (Schroedinger *et al.*, 2015). The creation of a Jalview



1	Name	Class
2	Q9FDB3_m2_spec_pro	pro
3	Q9R9J0_m4_spec_pro	pro
4	Q9RAH1_m2_spec_pro	pro
5	O05647_m1_spec_pro	pro
6	O30980_m1_spec_pro	pro

SSR-viz
The CSV_builder protocol can be used to create the csv annotation file, the SSR_plot protocol starts the plotting window, the Add_pdb protocol allows to add structure numbering to an '*.stats.csv' file

Input CSV file
Must be a csv file that with corresponding names to

Class label
An alternative column with the class label in the CSV

Alignment
Input file, must be a sequence alignment with

Verbose Mode
□ Shows additional details

Fig. 1. General flow chart of the SSR-viz toolbox. a) The input consists of a MSA together with a CSV-file holding the corresponding subfamily class labels. Multiple protein structures can also be included in order to assign the structure indices. b) The output can be created as a mathplotlib PDF, a Jalview annotation file or a summary CSV-File. This file is used to assign the indices of the structure. c) Example assignment of SSRs for the NRPS adenylation domain with specificity for valine and glycine.

(Clamp *et al.*, 2004) annotation file which allows to integrate the results in the Jalview Alignment Editor together with the original alignment.

2 Algorithm

Initially, each subfamily in the MSA is converted into a position probability matrix (PPM), which represents the probability to find a specific residue at a given position. This allows to compare the subfamilies independent of the number of their representing sequences. For each position (m) in the MSA the SSR score is computed based on the combination of three independent scoring functions (see Equation (1)).

$$SSR_{score}(m) = Conservation(m) * Difference(m) * \\ Exchange(m) \tag{1}$$

The conservedness $Conservation(m)$ of each position is defined by the normalized Shannon entropy of both subfamilies. The difference $Difference(m)$ between two subfamilies is computed by summation of an exchange matrix. The matrix represents the probability of each residue in one subfamily to be exchanged with another residue in the other subfamily. The more residues are exchanged the higher the score. This matrix judges each amino acid exchange equally. In order to differentiate the exchange of more divers amino acids, an additional scoring function $Exchange(m)$ is implemented. Again, the exchange matrix is computed, but this matrix can be weighed with a substitution matrix such as PAM or BLOSSUM. A purely physico-chemical substitution matrix is also available based on the research of Chrysostomou *et al.*, 2015. A detailed explanation of the algorithm including an hypothetical example is given in the Supplementary data.

3 Implementation

SSR-viz is based on a MSA as input, additionally a CSV-file is needed which supplies the subfamily specific label of each sequence. The CSV-File can be created using the **CSV_builder** tool. This tool allows also the label extraction from the sequence name using regular expressions. The CSV-file can have multiple labels, in case different types of functionality are investigated. The **CSV_builder** also excludes redundant sequences by default.

The detection of SSRs is handled by the **SSR_plot** tool. The detection threshold for SSRs can be assigned in two ways. Either the top N SSRs can be returned independent of their significance or the SSRs can be returned based on their Z-score, which only returns positions that have a significant higher score then the other positions in the alignment.

SSR-viz supports three different kind of outputs: A mathplotlib-style plot in PDF Format, which is customizable and consists of a heatmap, which visualizes all the scores and a plot which shows the most significant SSRs. A window function can also be added to the plot, which allows to investigate the alignment for areas of general importance.

It is often desired to visualize the SSRs together with the MSA, therefore, a Jalview annotation file can also be created, this file can be loaded into the Jalview Alignment Editor.

Finally, a summary CSV-file can be created, this file shows the SSRs together with the conserved positions in each subfamily class. In order to investigate the SSRs in a structural context another tool is implemented **Add_pdb**, which allows for mapping of protein structure indices from a PDB file to the indices in the MSA. **Add_pdb** supports any PDB file, downloaded from PDB or custom exported from PyMol, as long as it can be aligned to the MSA. A flow chart of the implementation is shown in Figure 1.

4 Use case

The toolbox was applied to detect SSRs for the adenylation domain of non-ribosomal peptide synthases (NRPS). The sequences and subfamily labels were extracted from the work of Rausch *et al.*, 2005. The input and generated output files are available in the GitHub project, including a detailed documentation which can help to reproduce the use case step-by-step.

Two subfamilies were chosen, with respective specificities for valine and glycine. For both subfamilies crystal structures are available (PDB: 3VNS, 4ZXI), so that the detected SSRs can be interpreted in a structural context. The 10 most important SSRs to distinguish the two subfamilies from each other were calculated (using the default parameters of SSR-viz). Additionally, the most important SSRs to distinguish all the subfamilies from each other (47 subfamiles, see supplementary data) were also calculated. All SSRs were subsequently mapped to the structure investigated by Stachelhaus *et al*., 1999 (PDB: 1AMU), which allows for a comparison with the described specificity-conferring code. All described residues are numbered based on the 1AMU sequence.

For the subfamilies with valine and glycine specificity all detected SSRs are in close proximity to the ligand (< 16 Angstrom, see supplementary Figure 1). The two most significant SSRs (236 and 331) flank the substrate binding side, which can explain the substrate choice due to sterical interaction as shown in Figure 1.

In the work of Stachelhaus *et al*., 1999, the residues are divided into highly variant and moderately variant residues. The highly variant residues show the highest flexibility with respect to substrate specificity. All the highly variant residues are detected by the all-vs-all scoring mode (239, 278, 299, 322, 331) in the top 10 SSRs.

For both scoring schemes only 5 SSRs overlap. Meaning, that they are equally important for distinguishing all the subfamilies from each other, as well as the two specifically observed families. This highlights, that subfamilies can be defined in more detail, by investigating SSRs independently for specific groups, then trying to establish an overall code for all subfamilies.

5 Conclusion

SSR-viz allows for a thorough analysis of protein subfamilies for SSRs based on different schemes and using various different substitution matrices. The diversity of the output options allows to further analyze the SSRs together with the alignment or in a structural context. We hope, that SSR-viz will improve the understanding of protein subfamily diversity by allowing the thorough analysis of the responsible SSRs.

Funding

This work has been supported by the German National Research Foundation [DFG, Research Training Group 1976].

References

Hannenhalli,S.S. and Russell,R.B. (2000) Analysis and prediction of functional subtypes from protein sequence alignments. *J. Mol. Biol.*, 303, 61-76.

Edwards,R.J. and Shields,D.C. (2005) BADASP: predicting functional specificity in protein families using ancestral sequences.*Bioinformatics*, 21, 4190-4191.

Olivera-Nappa,A. et al. (2011) Mutagenesis Objective Search and Selection Tool (MOSST): an algorithm to predict structure-function related mutations in proteins. *BMC Bioinform.*, 12, 122.

Suplatov,D. et al. (2014) Zebra: a web server for bioinformatic analysis of diverse protein families. *J. Biomol. Struct. Dyn.*, 32, 1752-1758.

Suplatov,D. et al. (2015) Robust enzyme design: bioinformatic tools for improved protein stability. *Biotechnol. J.*, 10, 344-355.

SchrÃ¶dinger, LLC (2015) The PyMOL Molecular Graphics System, Version 1.8.

Clamp,M. et al. (2004) The Jalview Java alignment editor. *Bioinformatics*, 20, 426-427.

Chrysostomou,C. and Seker,H. (2015) Novel protein weight matrix generated from amino acid indices. In, 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). 8181-8184.

Rausch,C. et al. (2005) Specificity prediction of adenylation domains in nonribosomal peptide synthetases (NRPS) using transductive support vector machines (TSVMs). *Nucleic Acids Res.*, 33, 5799-5808.

Stachelhaus,T. et al. (1999) The specificity-conferring code of adenylation domains in nonribosomal peptide synthetases. *Chem. Biol.*, 6, 493-505.