Bioinformatics, YYYY, 0–0
doi: 10.1093/bioinformatics/xxxxx
Advance Access Publication Date: DD Month YYYY
Applications Note

Structural bioinformatics

CapiPy: python based GUI-application to assist in protein immobilization

David Roura Padrosa[1,*], Valentina Marchini[1] and Francesca Paradisi[1]

[1]Department of Chemistry and Biochemistry, Freiestrasse 3. 3012, Bern, Switzerland.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Protein immobilization, while widespread to unlock enzyme potential in biocatalysis, remains tied to a trial an error approach. Nonetheless, several databases and computational methods have been developed for protein characterization and their study. CapiPy is a user-friendly application for protein model creation and subsequent analysis with a special focus on the ease of use and interpretation of the results to help the users to make an informed decision on the immobilization approach which should be ideal for a protein of interest. The package has been tested with three separate random sets of 150 protein sequences from Uniprot with more than a 70% overall success rate (see Supplementary information and Supplementary Dataset).

Availability and implementation: The package is free to use under the GNU General Public License v3.0. All necessary files can be downloaded from https://github.com/drou0302/CapiPy. All external requirements are also freely available, with some restrictions for non-academic users.

Contact: David.Roura@dcb.unibe.ch

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

1 Introduction

The use of enzymes to replace typical catalysts has attracted enormous attention as an alternative to traditional chemical synthesis (Sheldon & Woodley, 2018; Tamborini et al., 2018). Nonetheless, stabilization of these macromolecules is often needed to avoid the loss of catalytic activity. To overcome these limitations, immobilization has emerged as one of the most promising techniques for their stabilization and to allow their reuse (Bommarius & Paye, 2013; Romero-Fernández & Paradisi, 2020). Despite the great expectations and its widespread use, this technique is still very unpredictable, and each protein must be screened individually via a trial and error approach. This increased experimental effort, clearly impacts the real applicability of this technique. CapiPy, the package presented here, allows scientists with a know-how of protein immobilization, to exploits bioinformatics to rationalise their experimental design.

2 CapiPy description

CapiPy is a python based tool with graphical interface, based on Biopython (Cock et al., 2009) and PySimpleGUI (https://pysimplegui.readthedocs.io/). It can be installed and run in an Anaconda environment (https://www.anaconda.com/) or as a package from Python Package Index (https://pypi.org/). It is freely available, well documented and designed to be user-friendly incorporating executable files. The package is designed to be clear and easy to use for users with no experience python command line applications, but knowledge on protein immobilization. Certain functionalities are also provided as a stand-alone to maximize its flexibility. The codebase and installation have been tested in all newest versions of the main operating systems (Windows 10, macOS Catalina and Ubuntu 20.02 LTS).

Blast and Modelling

The script uses BLAST (Camacho et al., 2009), requiring the one letter code sequence of the query protein, to search against the Protein Data Bank database only. The blast result is stored, and the best hit is used as a template to create a model in MODELLER (Webb & Sali, 2014). If the template presents more than one chain in the original structure, the created model is cloned and superimposed to match the quaternary structure. The goodness of the final multimeric model is assessed with the RMSD compared to the template protein. The results are directly shown in PyMOL.

Active site identification

The second module, using either the results from the first one or a FASTA protein sequence as input, uses both UniProt (Bateman, 2019) and the M-CSA (Ribeiro et al., 2018) databases to identify the active site of the query protein. The best hit from UniProt or the subset of M-CSA with similar enzyme commission number (EC) is aligned using ClustalW2. To

avoid wrong numbering or inaccurate active site identification, if the identity between both sequences is lower than a 15%, no active site is yielded while if it falls between 15-40%, a warning is added to the results. The results are stored in a text file, which directly opens after execution along with the best hit from UniProt.

Surface and residue clustering

This script is designed, from a pdb formatted file, which can be the one created with the first module or specified by the user, calculate the minimum bounding box and volume of the protein (Guardado-Calvo, 2018) and the exposure of each residue using the half-sphere exposure measure (Heffernan et al., 2016; Song et al., 2008). Residues are classified in three categories: buried, semi-exposed or exposed. The exposed subset is then analyzed to find clusters (defined as a group of 3 or more residues with their Cα at less than 10Å) falling in one of 5 categories: positively charged, negatively charged, hydrophobic, lysine clusters or histidine clusters. These clusters, are saved in a PyMOL session (DeLano, 2002) file for easy visualization. Additionally, the possible interaction of any of this clusters to either the interchain interface, the active site residues or any user-specified residue can be assessed using Cluster distance module. If the specified residues are at less than 10 Å distance, a warning is printed into the file.

Immobilization literature retrieval

The last script included in the package combines the information from UniProt, by blasting the query sequence against the Swiss-Prot database (Bairoch & Apweiler, 2000), to retrieve the keywords to be searched in PubMed. The details of the 20 most relevant scientific publications are conveniently stores in a CSV file, compatible with common spreadsheet editors which again directly opens after execution.

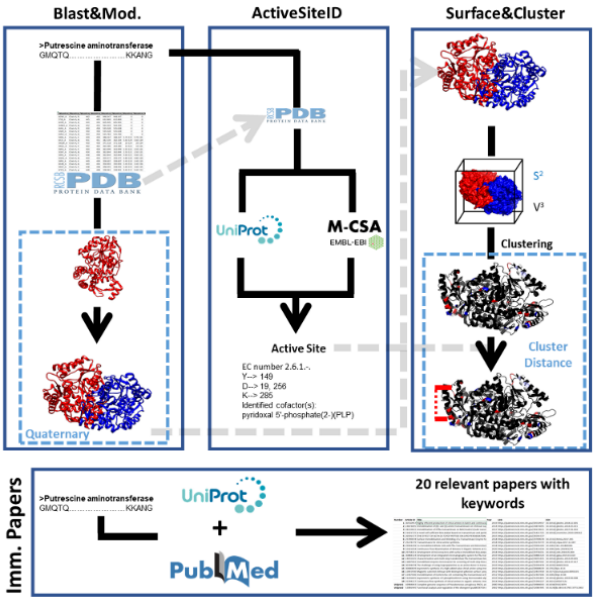

Figure 1: Scheme of CapiPy's workflow. The information and general pipeline of each module is detailed in the scheme above. Transparent grey lines indicate the information that can be shared between different scripts.

CapiPy performance

CapiPy has been tested with 3 sets of 150 randomly retrieved sequences from UniProt database with known EC number. In the retrieval, no bias depending on the size, quaternary assembly or EC code was detected (Supplementary information and Supplementary dataset). When run on CapiPy, less than 20% of the inputs resulted in code-related errors. The most common bottlenecks identified were the lack of high homology models either for the modelling, superposition or active site identification.

3 Conclusions

CapiPy, for its ease of installation, handling and result interpretation, constitutes a useful tool for scientists in the biocatalysis field with special focus on protein immobilization. Further work is now under development to expand CapiPy's functionality and compatibility with broadly used software, such as PyMOL or UCSF Chimera (Pettersen et al., 2004) as well as better linkage between the experimental data.

Acknowledgements

The authors wish to thank all members of the Paradisi Research group for their insights on the program design and implementation.

Funding

This work has been supported by the SNSF (200021_192274).

Conflict of Interest: none declared.

References

Bairoch, A., & Apweiler, R. (2000). The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Research*, *28*(1), 45–48. https://doi.org/10.1093/nar/28.1.45

Bateman, A. (2019). UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Research*, *47*(D1), D506–D515. https://doi.org/10.1093/nar/gky1049

Bommarius, A. S., & Paye, M. F. (2013). Stabilizing biocatalysts. *Chemical Society Reviews*, *42*(15), 6534. https://doi.org/10.1039/c3cs60137d

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., & Madden, T. L. (2009). BLAST+: architecture and applications. *BMC Bioinformatics*, *10*(1), 421. https://doi.org/10.1186/1471-2105-10-421

Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., & De Hoon, M. J. L. (2009). Biopython: Freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, *25*(11), 1422–1423. https://doi.org/10.1093/bioinformatics/btp163

DeLano, W. L. (2002). Pymol: An open-source molecular graphics tool. *{CCP4} Newsletter On Protein Crystallography*, *40*. http://www.ccp4.ac.uk/newsletters/newsletter40/11_pymol.pdf

Guardado-Calvo, P. (2018). *Python script to calculate and draw a minimal bounding box for a given protein. Version 2.* https://doi.org/10.13140/RG.2.2.32295.65446

Heffernan, R., Dehzangi, A., Lyons, J., Paliwal, K., Sharma, A., Wang, J., Sattar, A., Zhou, Y., & Yang, Y. (2016). Highly accurate sequence-based prediction of half-sphere exposures of amino acid residues in proteins. *Bioinformatics*, *32*(6), 843–849. https://doi.org/10.1093/bioinformatics/btv665

Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C., & Ferrin, T. E. (2004). UCSF Chimera?A visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, *25*(13), 1605–1612. https://doi.org/10.1002/jcc.20084

Ribeiro, A. J. M., Holliday, G. L., Furnham, N., Tyzack, J. D., Ferris, K., & Thornton, J. M. (2018). Mechanism and Catalytic Site Atlas (M-CSA): a database of enzyme reaction mechanisms and active sites. *Nucleic Acids Research*, *46*(D1), D618–D623. https://doi.org/10.1093/nar/gkx1012

Romero-Fernández, M., & Paradisi, F. (2020). Protein immobilization technology for flow biocatalysis. *Current Opinion in Chemical Biology*, *55*, 1–8. https://doi.org/10.1016/j.cbpa.2019.11.008

Sheldon, R. A., & Woodley, J. M. (2018). Role of Biocatalysis in Sustainable Chemistry. *Chemical Reviews*, *118*(2), 801–838. https://doi.org/10.1021/acs.chemrev.7b00203

Song, J., Tan, H., Takemoto, K., & Akutsu, T. (2008). HSEpred: predict half-sphere exposure from protein sequences. *Bioinformatics*, *24*(13), 1489–1497. https://doi.org/10.1093/bioinformatics/btn222

Tamborini, L., Fernandes, P., Paradisi, F., & Molinari, F. (2018). Flow Bioreactors as Complementary Tools for Biocatalytic Process Intensification. *Trends in Biotechnology*, *36*(1), 73–88. https://doi.org/10.1016/j.tibtech.2017.09.005

Webb, B., & Sali, A. (2014). Comparative Protein Structure Modeling Using MODELLER. *Current Protocols in Bioinformatics*, *47*(1), 5.6.1-5.6.32. https://doi.org/10.1002/0471250953.bi0506s47