




OpenCADD-KLIFS: A Python package to fetch kinase data from the KLIFS database

Dominique Sydow^{*1}, Jaime Rodríguez-Guerra¹, and Andrea Volkamer^{†1}

¹ *In Silico* Toxicology and Structural Bioinformatics, Institute of Physiology, Charité – Universitätsmedizin Berlin, corporate member of Freie Universität Berlin and Humboldt-Universität zu Berlin, Augustenburger Platz 1, 13353 Berlin, Germany

DOI: [10.21105/joss.03951](https://doi.org/10.21105/joss.03951)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: Richard Gowers 

Reviewers:

- [@ojeda-e](#)
- [@andrewtarzia](#)
- [@mcs07](#)

Submitted: 09 November 2021

Published: 14 February 2022

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

Summary

Protein kinases are involved in most aspects of cell life due to their role in signal transduction. Dysregulated kinases can cause severe diseases such as cancer, inflammation, and neurodegeneration, which has made them a frequent target in drug discovery for the last decades ([Cohen et al., 2021](#)). The immense research on kinases has led to an increasing amount of kinase resources ([Kooistra & Volkamer, 2017](#)). Among them is the KLIFS database, which focuses on storing and analyzing structural data on kinases and interacting ligands ([Kanev et al., 2020](#)). The OpenCADD-KLIFS Python module offers a convenient integration of the KLIFS data into workflows to facilitate computational kinase research.

[OpenCADD-KLIFS](#) (`opencadd.databases.klifs`) is part of the [OpenCADD](#) package, a collection of Python modules for structural cheminformatics.

Statement of need

The KLIFS resource ([Kanev et al., 2020](#)) contains information about kinases, structures, ligands, interaction fingerprints, and bioactivities. KLIFS thereby focuses especially on the ATP binding site, defined as a set of 85 residues and aligned across all structures using a multiple sequence alignment ([van Linden et al., 2014](#)). Fetching, filtering, and integrating the KLIFS content on a larger scale into Python-based pipelines is currently not straight-forward, especially for users without a background in online queries. Furthermore, switching between data queries from a *local* KLIFS download and the *remote* KLIFS database is not readily possible.

OpenCADD-KLIFS is aimed at current and future users of the KLIFS database who seek to integrate kinase resources into Python-based research projects. With OpenCADD-KLIFS, KLIFS data can be queried either locally from a KLIFS download or remotely from the KLIFS webserver. The presented module provides identical APIs for the remote and local queries and streamlines all output into standardized Pandas DataFrames ([The pandas development team, 2020](#)) to allow for easy and quick downstream data analyses ([Figure 1](#)). This Pandas-focused setup is ideal if you work with Jupyter notebooks ([Kluyver et al., 2016](#)).

^{*}corresponding author

[†]corresponding author

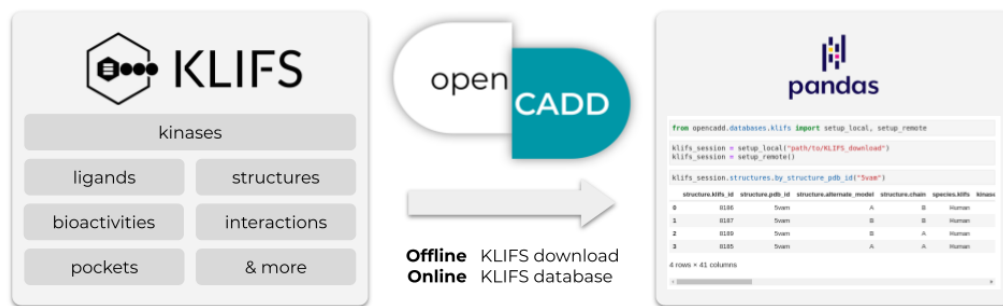


Figure 1: OpenCADD-KLIFS fetches KLIFS data (Kanev et al., 2020) offline from a local KLIFS download or online from the KLIFS database and formats the output as user-friendly Pandas DataFrames (The pandas development team, 2020).

State of the field

The KLIFS database is unique in the structure-based kinase field in terms of integrating and annotating different data resources in a kinase- and pocket-focused manner. Kinases, structures, and ligands have unique identifiers in KLIFS, which makes it possible to fetch and filter cross-referenced information for a query kinase, structure, or ligand.

- Kinase structures are fetched from the PDB, split by chains and alternate models, annotated with the KLIFS pocket of 85 residues, and aligned across the fully structurally covered kinome.
- Kinase-ligand interactions seen in experimental structures are annotated for the 85 pocket residues in the form of the KLIFS interaction fingerprint (KLIFS IFP).
- Bioactivity data measured against kinases are fetched from ChEMBL ([Mendez et al., 2018](#)) and linked to kinases, structures, and ligands available in KLIFS.
- Kinase inhibitor metadata are fetched from the PKIDB ([Carles et al., 2018](#)) and linked to co-crystallized ligands available in KLIFS.

The KLIFS data integrations and annotations can be accessed in different ways, which are all open source:

- Manually via the [KLIFS website](#) interface: This mode is preferable when searching for information on a specific structure or smaller set of structures.
- Automated via the [KLIFS KNIME](#) nodes ([Kooistra et al., 2018](#); [McGuire et al., 2017](#)): This mode is extremely useful if the users' projects are embedded in KNIME workflows; programming is not needed.
- Programmatically using the REST API and KLIFS OpenAPI specifications: This mode is needed for users who seek to perform larger scale queries or to integrate different queries into programmatic workflows. In the following, we will discuss this mode in context of Python-based projects and explain how OpenCADD-KLIFS improves the user experience.

The KLIFS database offers standardized URL schemes (REST API), which allows users to query data by defined URLs, using e.g. the Python package `requests` ([requests, 2021](#)). Instead of writing customized scripts to generate such KLIFS URLs, the KLIFS OpenAPI specifications — a document that defines the KLIFS REST API scheme — can be used to generate a Python client, using e.g. the Python package `bravado` ([bravado, 2021](#)). This client offers a Python API to send requests and receive responses. This setup is already extremely useful, however,

it has a few drawbacks: the setup is technical, the output is not easily readable for humans and not ready for immediate downstream integrations — requiring similar but not identical reformatting functions for different query results —, and switching from remote requests to local KLIFS download queries is not possible. Facilitating and streamlining these tasks is the purpose of OpenCADD-KLIFS as discussed in more detail in the next section.

Key Features

The KLIFS database offers a REST API compliant with the OpenAPI specification ([KLIFS, 2021](#)). Our module OpenCADD-KLIFS uses bravado to dynamically generate a Python client based on the OpenAPI definitions and adds wrappers to enable the following functionalities:

- A session is set up automatically, which allows access to various KLIFS *data sources* by different *identifiers* with the API `session.data_source.by_identifier`. *Data sources* currently include kinases, structures and annotated conformations, modified residues, pockets, ligands, drugs, and bioactivities; *identifiers* refer to kinase names, PDB IDs, KLIFS IDs, and more. For example, `session.structures.by_kinase_name` fetches information on all structures for a query kinase.
- The same API is used for local and remote sessions, i.e. interacting with data from a KLIFS download folder and from the KLIFS website, respectively.
- The returned data follows the same schema regardless of the session type (local/remote); all results obtained with bravado are formatted as Pandas DataFrames with standardized column names, data types, and handling of missing data.
- Files with the structural 3D coordinates deposited on KLIFS include full complexes or selections such as proteins, pockets, ligands, and more. These files can be downloaded to disc or loaded via biopandas ([Raschka, 2017](#)) or RDKit ([RDKit, 2021](#)).

OpenCADD-KLIFS is especially convenient whenever users are interested in multiple or more complex queries such as “fetching all structures for the kinase EGFR in the DFG-in conformation” or “fetching the measured bioactivity profiles for all ligands that are structurally resolved in complex with EGFR.” Formatting the output as DataFrames facilitates subsequent filtering steps and DataFrame merges in case multiple KLIFS datasets need to be combined.

OpenCADD-KLIFS is currently used in several projects from the Volkamer Lab ([Volkamer Lab, 2021](#)) including TeachOpenCADD ([TeachOpenCADD, 2021](#)), OpenCADD-pocket ([OpenCADD, 2021](#)), KiSSim ([KiSSim, 2021](#)), KinoML ([OpenKinome, 2021](#)), and PLIPify ([PLIPify, 2021](#)). For example, OpenCADD-KLIFS is applied in a [TeachOpenCADD tutorial](#) to demonstrate how to fetch all kinase-ligand interaction profiles for all available EGFR kinase structures to visualize the per-residue interaction types and frequencies with only a few lines of code.

Acknowledgements

We thank the whole KLIFS team for providing such a great kinase resource with an easy-to-use API and especially Albert Kooistra for his help with questions and wishes regarding the KLIFS database. We thank David Schaller for his feedback on the OpenCADD-KLIFS module. We acknowledge the contributors involved in software programs and packages used by OpenCADD-KLIFS, such as bravado, RDKit, Pandas, Jupyter, and Pytest, and Sphinx.

References

bravado. (2021). bravado. In *GitHub repository*. GitHub. <https://github.com/Yelp/bravado>

- Carles, F., Bourg, S., Meyer, C., & Bonnet, P. (2018). PKIDB: A Curated, Annotated and Updated Database of Protein Kinase Inhibitors in Clinical Trials. *Molecules*, 23(4), 908. <https://doi.org/10.3390/molecules23040908>
- Cohen, P., Cross, D., & Jänne, P. A. (2021). Kinase drug discovery 20 years after imatinib: Progress and future directions. *Nature Reviews Drug Discovery*, 20(7), 551–569. <https://doi.org/10.1038/s41573-021-00195-4>
- Kanev, G. K., de Graaf, C., Westerman, B. A., de Esch, I. J. P., & Kooistra, A. J. (2020). KLIFS: an overhaul after the first 5 years of supporting kinase research. *Nucleic Acids Research*, 49(D1), D562–D569. <https://doi.org/10.1093/nar/gkaa895>
- KiSSim. (2021). KiSSim: Subpocket-based fingerprint for kinase pocket comparison. In *GitHub repository*. GitHub. <https://github.com/volkamerlab/kissim>
- KLIFS. (2021). *KLIFS OpenAPI*. <https://dev.klifs.net>. https://dev.klifs.net/swagger_v2/
- Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., & team, J. development. (2016). Jupyter notebooks - a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Eds.), *Positioning and power in academic publishing: Players, agents and agendas* (pp. 87–90). IOS Press. <https://eprints.soton.ac.uk/403913/>
- Kooistra, A. J., Vass, M., McGuire, R., Leurs, R., Esch, I. J. P. de, Vriend, G., Verhoeven, S., & Graaf, C. de. (2018). 3D-e-Chem: Structural Cheminformatics Workflows for Computer-Aided Drug Discovery. *ChemMedChem*, 13(6), 614–626. <https://doi.org/10.1002/cmdc.201700754>
- Kooistra, A. J., & Volkamer, A. (2017). Chapter six - kinase-centric computational drug development. In R. A. Goodnow (Ed.), *Platform technologies in drug discovery and validation* (Vol. 50, pp. 197–236). Academic Press. <https://doi.org/10.1016/bs.armc.2017.08.001>
- McGuire, R., Verhoeven, S., Vass, M., Vriend, G., Esch, I. J. P. de, Lusher, S. J., Leurs, R., Ridder, L., Kooistra, A. J., Ritschel, T., & Graaf, C. de. (2017). 3D-e-chem-VM: Structural cheminformatics research infrastructure in a freely available virtual machine. *Journal of Chemical Information and Modeling*, 57(2), 115–121. <https://doi.org/10.1021/acs.jcim.6b00686>
- Mendez, D., Gaulton, A., Bento, A. P., Chambers, J., De Veij, M., Félix, E., Magariños, M. P., Mosquera, J. F., Mutowo, P., Nowotka, M., Gordillo-Marañón, M., Hunter, F., Junco, L., Mugumbate, G., Rodriguez-Lopez, M., Atkinson, F., Bosc, N., Radoux, C. J., Segura-Cabrera, A., ... Leach, A. R. (2018). ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Research*, 47(D1), D930–D940. <https://doi.org/10.1093/nar/gky1075>
- OpenCADD. (2021). OpenCADD-Pocket: Identification and analysis of protein (sub)pockets. In *GitHub repository*. GitHub. <https://github.com/volkamerlab/opencadd>
- OpenKinome. (2021). KinoML: Structure-informed machine learning for kinase modeling. In *GitHub repository*. GitHub. <https://github.com/openkinome/kinoml>
- PLIPify. (2021). PLIPify: Protein-ligand interaction frequencies across multiple structures. In *GitHub repository*. GitHub. <https://github.com/volkamerlab/plipify>
- Raschka, S. (2017). BioPandas: Working with molecular structures in pandas DataFrames. *The Journal of Open Source Software*, 2(14). <https://doi.org/10.21105/joss.00279>
- RDKit. (2021). RDKit: Open-Source Cheminformatics. In *RDKit website*. RDKit. <http://www.rdkit.org>
- requests. (2021). requests. In *GitHub repository*. GitHub. <https://github.com/psf/requests>

- TeachOpenCADD. (2021). TeachOpenCADD: a teaching platform for computer-aided drug design (CADD) using open source packages and data. In *GitHub repository*. GitHub. <https://github.com/volkamerlab/teachopencadd>
- The pandas development team. (2020). Pandas-dev/pandas: pandas. In *Zenodo repository*. Zenodo. <https://doi.org/10.5281/zenodo.3509134>
- van Linden, O. P. J., Kooistra, A. J., Leurs, R., Esch, I. J. P. de, & Graaf, C. de. (2014). KLIFS: A knowledge-based structural database to navigate kinase–ligand interaction space. *Journal of Medicinal Chemistry*, 57(2), 249–277. <https://doi.org/10.1021/jm400378w>
- Volkamer Lab. (2021). Volkamer Lab website. In *Volkamer Lab website*. Volkamer Lab. <https://volkamerlab.org/>