2023-03-24

volcalc: Calculate predicted volatility of chemical compounds

Kristina Riemer
Communications & Cybertechnologies, University of Arizona

Eric R. Scott
Communications & Cybertechnologies, University of Arizona

Laura Meredith
School of Natural Resources and the Environment, University of Arizona

Signatories

Project team

- Kristina Riemer, author/maintainer, Director of Communications and Cybertechnologies Data Science team at University of Arizona
- Eric Scott, contributor, Scientific Programmer and Educator for Communications and Cybertechnologies Data Science team at University of Arizona

Contributors

- Assistant Professor Laura Meredith developed the original idea for the `volcalc` package along with Kristina Riemer and is supportive of continued development by our team.
- PhD student S. Marshall Ledford has been the main user of early versions of `volcalc` and will continue to provide feedback on the package API and documentation.

Consulted

Tamás Stirling, maintainer of the `webchem` package (part of rOpenSci), was consulted and confirmed that `volcalc` is not replicating the efforts of any similar R packages that we are aware of.

The Problem

Volatile organic compounds (chemicals that readily evaporate under ambient conditions) are important in a number of fields and contexts including involvement in plant defense against herbivores, as routes of microbial communication, and as important indoor pollutants, to name a few. Yet measures of volatility are time consuming to calculate experimentally and not available for the vast majority of chemical compounds in chemical information databases. However, methods exist for predicting measures of volatility from chemical structure (Pankow & Asher, 2008). The `volcalc` package aims to automate the following steps for a given compound: 1) downloading data on chemical structure, 2) parsing those data to discover chemical functional groups, 3) applying the SIMPOL algorithm to predict volatility from functional groups and molecular weight. The current draft version of `volcalc` does all this, but is limited to working with compounds present in the Kyoto Encyclopedia of Genes and Genomes (KEGG) database (Kanehisa, 2000). This proposal focuses on expanding the scope of `volcalc` and preparing it for a larger and more diverse user base so anyone interested in volatile organic compounds can integrate it into their workflow.

The proposal

Overview

The current version of the main function in `volcalc`, `calc_vol()`, only works with compounds in the KEGG database. However, by refactoring existing code, we can make the volatility prediction functionality available to essentially any compound that has a known structure. Additionally, we plant to improve package infrastructure and documentation (i.e. tests, CI, vignettes, etc.), build the package on r-universe, and submit the package to CRAN in order to simplify installation. This will put a powerful and unique tool for calculating predicted volatility of essentially any compound in the hands of researchers in a wide variety of fields.

Detail

The current version of `volcalc` was created in 2022 as part of a data science incubator project in collaboration between Dr. Kristina Riemer and Dr. Laura Meredith at University of Arizona. `volcalc` is the first project, to our knowledge, to implement the SIMPOL method for predicting chemical vapor pressures and enthalpies of vaporization (Pankow & Asher, 2008) in an R package. This current version of the `volcalc` package has been successfully used to calculate

volatility estimates for *all* 16,000+ compounds in the Kyoto Encyclopedia of Genes and Genomes (KEGG) database. However, in its current form, it is limited to only working with chemical compounds in KEGG. The main function in `volcalc`, `calc_vol()`, currently downloads chemical information as molfiles from the KEGG API as a starting point. It then reads that .mol file in and parses it to find functional groups. It then applies an algorithm (published in Pankow and Asher (2008)) to predict vapor pressure, and outputs to the user a relative measure of volatility.

Molfile is an open format and various tools exist to translate other standard representations of chemical structure such as SMILES (Weininger, 1988) and InChI (Heller et al., 2013) to molfiles (e.g. using the `OpenBabel` command line utility or in R with `ChemmineOB`). Since `ChemmineOB` is already a dependency of `volcalc`, we can additionally extend `volcalc` to allow other representations of chemical structures as inputs. Since SMILES and InChl are both string representations, this will ideally make `volcalc` fit more easily into a `data.frame`-based workflow.

Refactoring the code in `volcalc` to work with essentially any chemical and preparing the package for wider distribution will make this powerful tool accessible to researchers across a variety of domains.

`volcalc` is developed on GitHub and distributed under an MIT license. Project repository: https://github.com/Meredith-Lab/volcalc

Project Goals

Our goals for the proposed project fall into two main categories: 1) to make `volcalc` useful for applications beyond estimating relative volatility for compounds in the KEGG database, and 2) polishing the package in preparation for an initial submission to CRAN.

For the first goal, we will focus on decoupling the data access and volatility estimation functionality of existing code in the `volcalc` package. This code is already written and the main work here is in API design and re-factoring. The minimum viable product here is a function that can calculate volatility when provided a path to a .mol file and a package vignette demonstrating how to couple this with chemical data sources such as the `webchem` package.

The second goal will involve improving test coverage (although current coverage is high at 96.88% as reported by `covr::package_coverage()`), establishing continuous integration with GitHub actions, improving documentation, and satisfying R CMD check. The minimum viable product here is a package that has gone through the steps suggested by `usethis::use_release_issue()` and is ready to submit to CRAN.

A reach goal is to allow users to input other chemical structure representations besides molfiles. The `ChemmineOB` package can translate a variety of chemical structure representations to molfiles and is already an indirect dependency of `volcalc` through its dependency on `ChemmineR`. Adding an argument to our `volcalc` function to specify the input format, and passing it to `ChemmineOB`'s translation function would be a way to expand the usability of `volcalc` even further.

Project plan

Dates

- Project start date: June 1, 2023
- Project end date: January 31, 2024

Start-up phase

I just realized there's nothing about getting package CRAN-ready on here

Milestone 1: July 1, 2023

- Implement CI with GitHub actions
- Check code coverage with `codecov` package
- Use GitHub Issues or Discussions to brainstorm eventual API (i.e. function names, argument names, how many exported functions, etc.)

Estimated work: 10 hours

Technical delivery

Milestone 2: September 1, 2023

- Re-factor `calc_vol()` code to split KEGG download and SIMPOL calculation functionality
- Deprecate arguments and functions appropriately as necessary
- Update documentation to reflect new function usage

Estimated work: 40 hours

Milestone 3: November 1, 2023

- Create a vignette demonstrating both KEGG usage and more general usage (i.e. providing a path to a .mol file) for volatility estimation
- Improve package documentation by adding citations, details, and additional examples where appropriate
- Create a `pkgdown` website for `volcalc`
- Create a CITATION.cff file, make a GitHub release, and archive code on Zenodo

Estimated work: 40 hours

Milestone 4: January 31, 2024

- Add functionality to supply other chemical representations besides molfiles as input
- Add to vignette(s) examples of integrating `volcalc` with data sources such as the `webchem` package to estimate volatility for an arbitrary set of compounds (i.e. not from KEGG)

Estimated work: 40 hours

Other aspects

Dissemination plan:

- After the initial re-factor (milestone 2 above), we plan to share the package with `webchem` contributors via our rOpenSci Slack channel for feedback & suggestions. We will encourage them to share the project with their networks of collaborators as well.
- At the project conclusion we will:
  - prepare blog post for https://datascience.cct.arizona.edu/
  - prepare a twitter announcement to share from **cct_datascience<empty citation>**
  - prepare a short demonstration video to be published to our YouTube channel that can be shared on the README of our repository and through social media
  - We will also work with our collaborator Laura Meredith to identify potential domain-specific venues to promote the use of `volcalc` such as scholarly societies, email lists, and social media accounts

Estimated work: 20 hours

Requirements

People

Kristina Riemer and Eric Scott will do the coding work for this project. Eric and Kristina both have experience creating R packages, writing tests, and collaborating using GitHub. Eric has experience submitting packages to CRAN and with automation using GitHub actions.

Eric and our collaborators Laura Meredith and S. Marshall Ledford have domain experience working with volatile organic compounds and chemoinformatics more generally. S. Marshall Ledford is currently the sole user of `volcalc` besides the developers, and is interested in giving feedback on future versions.

Funding

We would like to request funding for the salaries of personnel working on this project.

- Kristina Riemer: 75 hours, $80/hr, altogether $6,000
- Eric Scott: 75 hours, $80/hr, altogether $6,000

For a total of $12,000
Funding timeline:

- After milestone 2 (September 1, 2023), $6,000
- After technical delivery (January 31, 2024), $6,000

Summary

These costs represent a one-time investment to pay for personnel to work on the project.

Success

Definition of done

We would consider this project successful when a new version of `volcalc` with the ability to calculate volatility given an arbitrary molfile has been released on GitHub, the code has been archived on Zenodo, and the package has been successfully submitted to CRAN.

Measuring success

The following can be used to measure success:

- Can the package be installed from GitHub? (yes/no)
- Can the package be installed from r-universe? (yes/no)
- Test coverage (at least 90%)
- Correctness tests for volatiltiy predictions including comparisons with measured values (± some tolerance) (yes/no)
- Is the package passing R CMD check on linux, macOS, and windows using CI? (yes/no)
- Has the package been tested by users other than the developers? (yes/no)
- Does the package have a vignette that is easy to follow? (yes/no)
- Is code archived on Zenodo and a DOI associated with the package citation? (yes/no)
- The number of chemical representation types that can be used as input (molfile for success; inchikey, inchi, SMILES, and possibly more for reach goal)

Future work

- visualizations?
- present at a conferences or teaching people to use it
- what publications could this enable

Key risks

- The project could be delayed if contributors ended up able to devote less time to the project than planned
- The SIMPOL algorithm might not be applicable to *all* compounds, since it was designed to work with volatile organic compounds. In that case, we may need to add a warning to the user that values returned by `volcalc` may not make sense for certain compounds.

- `volcalc` has `OpenBabel` as a system dependency (indirectly), which could potentially lead to delays in getting the package to build using GitHub actions.

References

Heller, S., McNaught, A., Stein, S., Tchekhovskoi, D., & Pletnev, I. (2013). Inchi - the worldwide chemical structure identifier standard. *Journal of Cheminformatics*, *5*(1). https://doi.org/10.1186/1758-2946-5-7

Kanehisa, M. (2000). Kegg: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, *28*(1), 27–30. https://doi.org/10.1093/nar/28.1.27

Pankow, J. F., & Asher, W. E. (2008). Simpol.1: A simple group contribution method for predicting vapor pressures and enthalpies of vaporization of multifunctional organic compounds [Citation Key: pankowSIMPOLSimpleGroup2008]. *Atmos. Chem. Phys.* https://doi.org/10.5194/acp-8-2773-2008

Weininger, D. (1988). Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, *28*(1), 31–36.