

Reproducibility Review

Below is a seven point reproducibility review prescribed by Improving reproducibility and reusability in the Journal of Cheminformatics of the **main** branch of repository https://github.com/jonghyunlee1993/DLM-DTI_hint-based-learning (commit **bf267706**), accessed on 2023-08-29.

Criteria

Does the repository contain a **LICENSE** file in its root?

No,

the GitHub license picker can be used to facilitate adding one by following this link: https://github.com/jonghyunlee1993/DLM-DTI_hint-based-learning/community/license/new?branch=main.

Ideal software licenses for open source software include the MIT License, BSD-3 Clause License, and other licenses approved by the Open Source Initiative. A simple, informative guide for picking a license can be found at <https://choosealicense.com>.

More information about how GitHub detects licenses can be found [here](#).

Does the repository contain a **README** file in its root?

Yes.

Does the repository contain an associated public issue tracker?

Yes.

Has the repository been externally archived on Zenodo, FigShare, or equivalent that is referenced in the **README**?

No,

this repository has a README, but it does not reference Zenodo. If your Zenodo record is XYZ, then you can use the following in your README:

 [\[! DOI\] \(https://zenodo.org/badge/DOI/10.5281/zenodo.XYZ.svg\)](https://zenodo.org/badge/DOI/10.5281/zenodo.XYZ.svg) (<https://doi.org/10.5281/zenodo.XYZ>)

Does the **README** contain installation documentation?

No,

this repository has a README, but it does not contain a section header entitled **# Installation** (it's allowed to be any level deep). Please add a section that includes information on how the user should get the code (e.g., clone it from GitHub) and install it locally. This might read like:

```
git clone https://github.com/jonghyunlee1993/DLM-DTI_hint-based-learning
cd DLM-DTI_hint-based-learning
pip install --editable .
```

Alternatively, you can deploy your code to the Python Package Index (PyPI) and document how it can be installed with `pip install`. This might read like:

```
pip install dlm_dti_hint_based_learning
```

Is the code from the repository installable in a straight-forward manner?

No,

no packing setup configuration (e.g., `setup.py`, `setup.cfg`, `pyproject.toml`) was found. This likely means that the project can not be installed in a straight-forward, reproducible way. Your code should be laid out in a standard structure and configured for installation with one of these files. See the following resources:

- <https://packaging.python.org/en/latest/tutorials/packaging-projects/>
- <https://blog.ionelm.ro/2014/05/25/python-packaging>
- <https://hynek.me/articles/testing-packaging/>
- <https://cthoit.com/2020/06/03/how-to-code-with-me-organization.html>

Note that the following do not qualify as straightforward and reproducible because their goals are to set up an environment in a certain way, and not to package code such that it can be distributed and reused.

1. `requirements.txt`
2. Conda/Anaconda environment configuration

Does the code conform to an external linter (e.g., `black` for Python)?

No,

the repository does not conform to an external linter. This is important because there is a large cognitive burden for reading code that does not conform to community standards. Linters take care of formatting code to reduce burden on readers, therefore better communicating your work to readers.

For example, `black` can be applied to auto-format Python code with the following:

```
git clone https://github.com/jonghyunlee1993/DLM-DTI_hint-based-learning tmp
cd tmp
python -m pip install black
black .
git commit -m "Blacken code"
git push
```

Summary

Scientific integrity depends on enabling others to understand the methodology (written as computer code) and reproduce the results generated from it. This reproducibility review reflects steps towards this goal that may be new for some researchers, but will ultimately raise standards across our community and lead to better science.