The Journal of Open Source Software

Bedparse: feature extraction from BED files

Tommaso Leonardi[1, 2]

1 The Gurdon Institute, University of Cambridge, Cambridge, UK **2** Center for Genomic Science IIT@SEMM, Istituto Italiano di Tecnologia (IIT), Milan, Italy

DOI: 10.21105/joss.01228

Software

- Review
- Repository
- Archive

Submitted: 22 January 2019
Published: 28 February 2019

Summary

`Bedparse` is a Python module and command-line interface tool to extract features from genome annotation files in BED (Browser Extensible Data) format. The BED format is a plaintext file format commonly used in bioinformatics to represent genomic features. Each line in a BED file corresponds to a genomic feature (e.g. a gene, transcript, peak, regulatory region, etc.) and consists of up to 12 tab-separated fields that define its genomic coordinates and exon-intron structure. This format is also commonly used to graphically visualise genomic features by genome browser software and is one of the standard formats used by the UCSC (Kent et al., 2002) and Ensembl (Zerbino et al., 2018) genome browsers. One of the major advantages of the BED format over many of its alternatives is that each line includes all the information required to define an individual gene/transcript model. This makes the format particulary convenient when used with Unix pipes, `awk` one-liners or small custom scripts. This ad hoc approach, albeit (usually) simple and effective, often leads to repetition and/or code duplication and can be prone to errors, bugs and typos that are not always easy to detect.

`Bedparse` aims to simplify and standardise many of the operations and feature extractions commonly done on BED files by adhering to the Unix philosophy of doing one thing and doing it well (Prins, 2014). Despite the simplicity of many of its functions, `bedparse` is thoroughly and rigorously tested through an automated test suit to ensure the accuracy and correctness of the results. Additionally, `bedparse` performs syntax validation checks on the input BED files and warns the user in case of inconsistencies or malformed files.

`Bedparse` implements the following functions:

- Filtering of transcripts based on annotations
- Joining of annotation files based on transcript names
- Promoter reporting
- Intron reporting
- CDS reporting
- UTR reporting

In addition to the feature-extraction functions reported above, `bedparse` also provides three format conversion tools:

- `convertChr` implements an internal dictionary that allows conversion of human and mouse chromosome names (including patches) between the two most widely used formats, i.e. the Ensembl and the UCSC naming schemes.
- `gtf2bed` allows converting Ensembl/Gencode Gene Transfer Format (GTF) files into BED format, with options to specify extra fields to add after column 12.

- `bed12tobed6` converts BED12 files to the BED6 format.

Internally, `bedparse` implements a `bedline` class that performs several checks on each BED field in order to ensure the correctness of the format and implements methods that perform the functions listed above. This design allows `bedparse` to be either imported by other projects as a Python module or as a standalone tool through its command-line interface, either on its own or as part of a *pipe*.

In conclusion, `bedparse` is a light, versatile and portable tool developed using good programming practices and a test-driven development approach. Its use as part of bioinformatics pipelines will contribute to speeding up development time and preventing bugs.

`Bedparse` is open source and released under the MIT Licence. The source code is hosted on Github, and releases are automatically tested using Travis CI and archived on Zenodo.

Acknowledgments

The author would like to thank Adrien Leger and Luca Pandolfini for testing `bedparse` and giving useful feedback, and Tony Kouzarides for providing funding for the author.

References

Kent, W. J., Sugnet, C. W., Furey, T. S., Roskin, K. M., Pringle, T. H., Zahler, A. M., Haussler, et al. (2002). The human genome browser at ucsc. *Genome Research*, *12*(6), 996–1006. doi:10.1101/gr.229102

Prins, P. (2014). The small tools manifesto for bioinformatics. doi:10.5281/zenodo.11321

Zerbino, D. R., Achuthan, P., Akanni, W., Amode, M. R., Barrell, D., Bhai, J., Billis, K., et al. (2018). Ensembl 2018. *Nucleic Acids Research*, *46*(D1), D754–D761. doi:10.1093/nar/gkx1098