

Sit2StandPy: An Open-Source Python Package for Detecting and Quantifying Sit-to-Stand Transitions Using an Accelerometer on the Lower Back

Lukas Adamowicz¹ and Shyamal Patel¹

¹ Pfizer, Inc. 610 Main Street, Cambridge MA, USA 02139

DOI: [10.21105/joss.02449](https://doi.org/10.21105/joss.02449)

Software

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

Editor: [Gabriela Alessio Robles](#) ↗

Reviewers:

- [@jsgalan](#)
- [@gweissman](#)

Submitted: 23 June 2020

Published: 11 August 2020

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

Sit-to-stand transitions have been used to assess mobility for a broad range of conditions, such as Parkinson's Disease (Buckley, Pitsikoulis, Barthelemy, & Hass, 2009) and knee osteoarthritis (Bolink, van Laarhoven, Lipperts, Heyligers, & Grimm, 2012). Assessments are typically performed in the clinic using timed performance tests like the timed-up-and-go (Nguyen et al., 2015, 2017) and chair stand tests (Guralnik, 1994). While these assessment tools have demonstrated good psychometric properties, they have two key limitations: (1) assessments are performed episodically as they need to be administered by trained examiners, and (2) assessments performed in the clinic might not provide an adequate measure of real-world mobility. Therefore, there is a growing interest (Martinez-Hernandez & Dehghani-Sanij, 2019; Pham, Warmerdam, Elshehabi, Schlenstedt, & Bergeest, 2018) in the use of wearable devices to detect sit-to-stand transitions that occur during daily life and quantify the quality of mobility during these transitions. While some commercial wearable sensor systems have sit-to-stand algorithms (e.g. APDM), and some previous papers have released code (Martinez-Hernandez & Dehghani-Sanij, 2019), to the authors' knowledge there are no commonly used, open source, and easy to use and configure, implementations of sit-to-stand algorithms available.

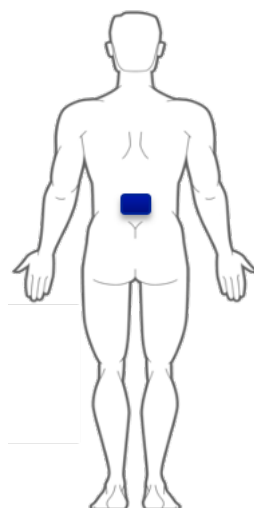


Figure 1: Location of the wearable device on the lower back.

Sit2StandPy is an open source Python package that implements novel heuristics-based algorithms to detect Sit-to-Stand transitions from accelerometer data captured using a single

wearable device mounted on the lower back (Figure 1). For each detected transition, the package also calculates objective features to assess quality of the transition. Features include duration, maximum acceleration, minimum acceleration, and spectral arc length (SPARC) (Balasubramanian, Melendez-Calderon, Roby-Brami, & Burdet, 2015). While most previous works have focused on either in-clinic applications (Nguyen et al., 2015, 2017; van Lummel et al., 2013) or the use of multiple sensors (Nguyen et al., 2015, 2017), the algorithms in this package can handle data collected under free-living conditions as well as prescribed tasks (e.g. 30-second chair stand task), and it uses only the acceleration data from a single sensor on the lower back with unconstrained device orientation. The practicality of a single-accelerometer approach, which affords a long battery life and improved wearability, makes it well suited for long-term, continuous monitoring at home.

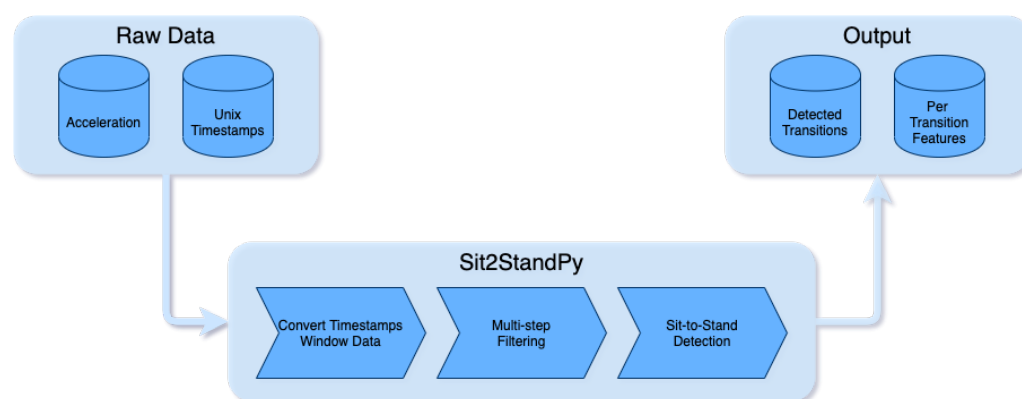


Figure 2: A high-level illustration of the input, output, and main processing steps of Sit2StandPy.

As illustrated in Figure 2, Sit2StandPy takes raw accelerometer data with timestamps as input and returns detected sit-to-stand transitions. Data can be windowed by full days, or parts of days can be selected for each window (e.g. window from 08:00 to 20:00). A high-level interface is provided, which allows the user to access all adjustable parameters. Users provide raw data as input and get detected transitions along with the computed features as output. Additionally, the lower-level methods that are called during the detection are available as well for more fine-grained control.

With this framework, users can control many parameters of transition detection. In addition, the separation of processing steps and modularity of the sub-processes allows for easy customization if desired. For example, users can easily customize functions for generating additional features to assess quality of a transition.

Availability

Sit2StandPy is distributed under the MIT License and is published on PyPI, the Python Package Index, and can be installed by running the following in the terminal:

```

pip install sit2standpy # install with checking for dependencies
# or
# installation without checking for installed dependencies
pip install sit2standpy --no-deps
  
```

Sit2StandPy requires the following Python packages:

- Python ≥ 3.7

- NumPy - (van der Walt, Colbert, & Varoquaux, 2011)
- SciPy - (Jones, Oliphant, & Peterson, 2001)
- pandas - (The pandas development team, 2020)
- PyWavelets - (Lee, Gommers, Wasilewsky, Wohlfahrt, & O’Leary, 2019)
- udatetime

Sit2StandPy contains example code with sample data in its GitHub repository. Full documentation with usage examples, installation instructions, and API reference are all available at <https://sit2standpy.readthedocs.io/en/latest/>

Acknowledgements

The Digital Medicine & Translational Imaging group at Pfizer, Inc supported the development of this package.

References

- Balasubramanian, S., Melendez-Calderon, A., Roby-Brami, A., & Burdet, E. (2015). On the analysis of movement smoothness. *Journal of NeuroEngineering Rehabilitation*, 12(1). doi:[10.1186/s12984-015-0090-9](https://doi.org/10.1186/s12984-015-0090-9)
- Bolink, S. A. A. N., van Laarhoven, S. N., Lipperts, M., Heyligers, I. C., & Grimm, B. (2012). Inertial sensor motion analysis of gait, sit–stand transfers and step-up transfers: Differentiating knee patients from healthy controls. *Physiological Measurement*, 33(11), 1947–1958. doi:[10.1088/0967-3334/33/11/1947](https://doi.org/10.1088/0967-3334/33/11/1947)
- Buckley, T., Pitsikoulis, C., Barthelemy, E., & Hass, C. J. (2009). Age impairs sit-to-walk motor performance. *Journal of Biomechanics*, 42(14), 2318–2322. doi:[10.1016/j.jbiomech.2009.06.023](https://doi.org/10.1016/j.jbiomech.2009.06.023)
- Guralnik, et a., J. M. (1994). A short physical performance battery assessing lower extremity function: Association with self-reported disability and prediction of mortality and nursing home admission. *Journal of Gerontology*, 49(2), 85–94. doi:[10.1093/geronj/49.2.m85](https://doi.org/10.1093/geronj/49.2.m85)
- Jones, E., Oliphant, T., & Peterson, et a., P. (2001). SciPy: Open source scientific tools for Python. Retrieved from <http://www.scipy.org/>
- Lee, G. R., Gommers, R., Wasilewsky, F., Wohlfahrt, K., & O’Leary, A. (2019). PyWavelets: A Python package for wavelet analysis. *Journal of Open Source Software*. doi:[10.21105/joss.01237](https://doi.org/10.21105/joss.01237)
- Martinez-Hernandez, U., & Dehghani-Sanij, A. A. (2019). Probabilistic identification of sit-to-stand and stand-to-sit with a wearable sensor. *Pattern Recognition Letters*, 118. doi:[10.1016/j.patrec.2018.03.020](https://doi.org/10.1016/j.patrec.2018.03.020)
- Nguyen, H. P., Ayachi, F., Lavigne–Pelletier, C., Blamoutier, M., Rahimi, F., Boissy, P., Mandar, J., et al. (2015). Auto detection and segmentation of physical activities during a timed-up-and-go (TUG) task in healthy older adults using multiple inertial sensors. *Journal of NeuroEngineering and Rehabilitation*, 12(1), 36. doi:[10.1186/s12984-015-0026-4](https://doi.org/10.1186/s12984-015-0026-4)
- Nguyen, H. P., Lebel, K., Boissy, P., Bogard, S., Goubault, E., & Duval, C. (2017). Auto detection and segmentation of daily living activities during a timed up and go task in people with parkinson’s disease using multiple inertial sensors. *Journal of NeuroEngineering and Rehabilitation*, 14(1), 26. doi:[10.1186/s12984-017-0241-2](https://doi.org/10.1186/s12984-017-0241-2)

- Pham, M. H., Warmerdam, E., Elshehabi, M., Schlenstedt, C., & Bergeest, L. e. a. (2018). Validation of a lower back “wearable”-based sit-to-stand and stand-to-sit algorithm for patients with parkinson’s disease and older adults in a home-like environment. *Frontiers in Neurology*, 9:652. doi:[10.3389/fneur.2018.00652](https://doi.org/10.3389/fneur.2018.00652)
- The pandas development team. (2020). *Pandas-dev/pandas: Pandas 1.0.5*. Zenodo. doi:[10.5281/zenodo.3898987](https://doi.org/10.5281/zenodo.3898987)
- van der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The numpy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2), 22–30. doi:[10.1109/MCSE.2011.37](https://doi.org/10.1109/MCSE.2011.37)
- van Lummel, R. C., Ainsworth, E., Lindemann, U., Zijlstra, W., Chiari, L., Van Campen, P., & Hausdorff, J. M. (2013). Automated approach for quantifying the repeated sit-to-stand using one body fixed sensor in young and older adults. *Gait & Posture*, 38(1), 153–156. doi:[10.1016/j.gaitpost.2012.10.008](https://doi.org/10.1016/j.gaitpost.2012.10.008)