

# pyPhenology: A python framework for plant phenology modelling

Shawn David Taylor<sup>1</sup>

DOI: [10.21105/joss.00827](https://doi.org/10.21105/joss.00827)

<sup>1</sup> School of Natural Resources and Environment, University of Florida

## Software

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

Submitted: 04 July 2018

Published: 15 August 2018

## License

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

## Summary

Phenology, the timing of biological events, is an important aspect of environmental systems and phenology models have been in use since the mid 20th century (Chuine & Régnière, 2017). Phenology is a well established field of research and there are numerous model comparison studies which attempt to find the model which best explains a specific phenological event (Basler, 2016; Tang et al., 2016). Many phenology models are well established (Chuine, de Cortazar-Atauri, Kramer, & Hänninen, 2013), yet studies frequently implement a custom codebase to evaluate them. The pyPhenology package attempts to create a common modelling framework which can be used and referenced in phenology research. This will allow other researchers to check the implementation of mathematical models and add new models without rewriting core functionality.

pyPhenology has an object oriented API where the same analysis code can be used regardless of the underlying model. The API is inspired by scikit-learn, having `fit()` and `predict()` methods for all models (Pedregosa et al., 2011). This allows for easy and reproducible code for phenology model selection and comparison studies. The package implements model fitting using built-in optimizers from the `scipy` package, allowing end-users to fit models to their data with just a few lines of code (Jones, Oliphant, Peterson, & Others, 2001). A common phenology model requirement is setting certain parameters to fixed values, for example setting the first day for degree day accumulation to January 1. This is done via a simple argument in model initialization. Fitted models can be saved for later use or model parameters exported as a python dictionary for inclusion other analysis. New models can be easily added and must only implement the actual model equations and a checks for adequate explanatory variables. All other requirements, such as fitting, predictions, or model saving, can be inherited from a parent class.

The package comes preloaded with several phenology datasets for users to test and learn with. The first is flower and budburst observations of highbush blueberry (*Vaccinium corymbosum*) from Harvard Forest (O'Keefe, 2015). The second is flower, budburst, and fall senescence observations of aspen (*Populus tremuloides*) from the U.S.A. National Phenology Network and the many participants who contribute to its Nature's Notebook program (USA National Phenology Network, 2017). Daily mean temperature for all locations represented in the data was obtained from the PRISM dataset (PRISM Climate Group, 2004).

pyPhenology was built with large scale analysis in mind and currently drives the continental scale phenology models on <http://phenology.naturecast.org>. A similar phenology modeling framework, `phenor`, is currently available for the R language (Hufkens, Basler, Milliman, Melaas, & Richardson, 2018).

## Acknowledgements

Development of this software was funded by the Gordon and Betty Moore Foundation's Data-Driven Discovery Initiative through Grant GBMF4563 to Ethan P. White.

## References

- Basler, D. (2016). Evaluating phenological models for the prediction of leaf-out dates in six temperate tree species across central Europe. *Agricultural and Forest Meteorology*, 217, 10–21. doi:[10.1016/j.agrformet.2015.11.007](https://doi.org/10.1016/j.agrformet.2015.11.007)
- Chuine, I., & Régnière, J. (2017). Process-Based Models of Phenology for Plants and Animals. *Annu. Rev. Ecol. Evol. Syst.*, 48, 159–82. doi:[10.1146/annurev-ecolsys-110316-022706](https://doi.org/10.1146/annurev-ecolsys-110316-022706)
- Chuine, I., de Cortazar-Atauri, I. G., Kramer, K., & Hänninen, H. (2013). Plant Development Models. In M. D. Schwartz (Ed.), *Phenology: An Integrative Environmental Science* (pp. 275–293). Dordrecht: Springer Netherlands. doi:[10.1007/978-94-007-6925-0\\_15](https://doi.org/10.1007/978-94-007-6925-0_15)
- Hufkens, K., Basler, D., Milliman, T., Melaas, E. K., & Richardson, A. D. (2018). An integrated phenology modelling framework in R. (S. Goslee, Ed.) *Methods in Ecology and Evolution*, 9(5), 1276–1285. doi:[10.1111/2041-210X.12970](https://doi.org/10.1111/2041-210X.12970)
- Jones, E., Oliphant, T., Peterson, P., & Others. (2001). SciPy: Open Source Scientific Tools for Python.
- O’Keefe, J. (2015). Phenology of woody species at harvard forest since 1990. Harvard forest data archive: HF003. Data set accessed 05/01/2017 at <http://harvardforest.fas.harvard.edu:8080/exist/apps/datasets/showData.html?id=hf003>.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- PRISM Climate Group. (2004). <http://prism.oregonstate.edu>; Oregon State University.
- Tang, J., Körner, C., Muraoka, H., Piao, S., Shen, M., Thackeray, S. J., & Yang, X. (2016). Emerging opportunities and challenges in phenology: A review. *Ecosphere*, 7(8), e01436. doi:[10.1002/ecs2.1436](https://doi.org/10.1002/ecs2.1436)
- USA National Phenology Network. (2017). Plant and animal phenology data. Data type: Status and intensity. 01/01/2009-04/31/2017 for region: 49.9375°, -66.4791667° (ur); 24.0625°, -125.0208333° (ll). USA-NPN, Tucson, Arizona, USA. Data set accessed 05/01/2017 at <http://doi.org/10.5066/F78S4N1V>.