

# HARMONIZING MACHINE LEARNING BASED PHENOLOGICAL MODELING: A UNIFIED WORKFLOW FOR COMPARATIVE ANALYSES

Mahdi Khodadadzadeh<sup>1</sup>, Peter Kalverla<sup>2</sup>, Raúl Zurita-Milla<sup>1</sup>

<sup>1</sup>Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente,  
PO Box 217, 7500 AE, Enschede, the Netherlands

<sup>2</sup> Netherlands eScience Center, Science Park 140, 1098 XG Amsterdam, the Netherlands

## ABSTRACT

Undoubtedly, climate change is influencing plant distribution and growth. Phenology, the study of events such as leafing and flowering timing, plays a crucial role in understanding and monitoring climate change impacts. Phenological modeling stands as a vital tool to quantify the influence of environmental changes on plants. Nevertheless, a lack of standardization among phenology prediction models hinders comparisons between different approaches. This paper introduces a workflow utilizing existing data, tools and software packages, thereby easing comparisons among various process-based phenology prediction models and machine learning methods (including classification and regression). The primary objective of this article is to promote consistency in phenological modeling and model evaluation.

**Index Terms**— Climate change, plant phenology, process-based models, machine learning models

## 1. INTRODUCTION

Phenology is the scientific discipline that explores the timing of plant life-cycle phases, such as flowering and leafing, the factors behind these timings, and the connections between growth stages of similar or dissimilar species [1]. Phenological studies are crucial for understanding the interplay between plants' reactions and environmental and climatic variables, as well as for executing strategies to mitigate and adapt to climate change.

Data and models are two pillars of phenological studies. Multiple observational networks (such as the USA National Phenology Network (USA-NPN) [2] and the Pan-European Phenology Database (PEP725) [3]) offer national- and continent-scale phenological datasets that are readily accessible. In addition, other data sources are employed to incorporate environmental variables that affect phenology, such as temperature, precipitation, and air humidity. Remote sensing sensors, both near and far, are also often used to produce phenology related data products. This diversity of data sources complicates integration and particularly data preparation for modeling [4].

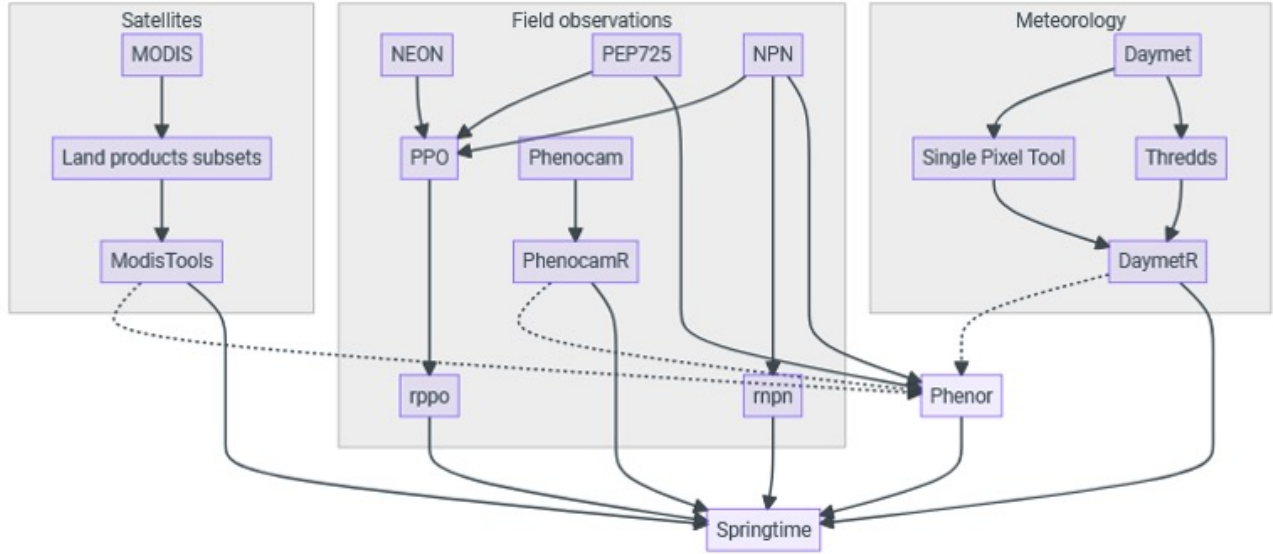
In the realm of modeling, two prominent and seemingly divergent trends in phenological modeling involve process-based models utilizing metrics like growing degree days (GDD) or growing season index (GSI), and data-driven machine learning (ML) models [5]. Despite the presence of software packages like pyPhenology [6] and Phenor [7], aiming to establish a common foundation for phenological modeling, **there is currently no standardized workflow specification that encompasses both process-based and advanced ML-based models**. Phenological research, in particular, calls for a flexible modeling framework that can seamlessly integrate advanced ML techniques, such as mixed effects models [8] and state-of-the-art interpretable ML models [9]. Such a unified framework holds the promise of delivering precise, reliable, comprehensible, and comparable predictions, serving as a reference in phenological studies. This work represents one of the initial efforts in harmonizing phenological modeling.

## 2. METHODS

This paper relies on the Springtime Python package [10], an open-source tool specifically designed for plant phenological research, emphasizing the integration of phenological data and models. Springtime supports researchers by facilitating access, pre-processing, and integration of phenological data, along with the implementation of modeling using a broad spectrum of process-based and advanced ML models. The software places significant emphasis on streamlining both data and modeling workflows, allowing users to execute them seamlessly with a single command. It adopts standardized locations for storing raw and intermediate data and utilizes a standardized *recipe* format to define the sequential steps in the workflows. Springtime executes ML algorithms that are supported by the package PyCaret [11] and process-based models from **pyPhenology** [6].

### 2.1. Data

As depicted in Fig. 1, Springtime consolidates data sets from various sources. In this study, we create a modeling-ready



**Fig. 1:** Springtime's data sources and existing tools for data preparation

dataset for predicting the emergence of the first leaf in common lilac (*Syringa vulgaris*) throughout the United States. The dataset is constructed by combining observations from the NPN network spanning 2000 to 2021, integrated with daily mean temperature data from the Daymet dataset.

## 2.2. Process-based phenological modeling

Process-based models aim to simulate underlying biological, physical, and chemical processes. They are designed to capture the mechanisms controlling the timing of various events, such as the emergence of the first leaf. For example, the Thermal Time (TT) model, one of the earliest models developed, assumes that leaf emergence occurs on the day the accumulated rate of daily heating temperatures surpasses a specific threshold. An event happens on the day of the year (*DOY*) when the following is met:

$$\sum_{t=t_1}^{DOY} R_f(T_i) \geq F \quad (1)$$

where,

$$R_f(T_i) = \max(T_i - T_{base}, 0) \quad (2)$$

As you can see, the TT model can be built by setting three parameters:  $t_1$  the start of the accumulation (i.e., the *DOY* which forcing accumulating beings),  $T_{base}$  the threshold temperature above which forcing accumulates and  $F$  is the total forcing units required.

## 2.3. Machine learning-based phenological modeling

Beyond the phenology community, numerous advanced techniques have emerged in the field of ML in recent years. There

is a growing trend towards the development of ML-based phenological models. For example in [12], 18 different ML models including Random Forest (RF), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost) are tested and compared to predict leaf emergence. Although ML algorithms have been largely explored, the formulation of ML-based phenological modeling has not been well-defined in a way that facilitates easy application with existing tools. In the following, we elaborate on how phenological modeling can be conceptualized as either a binary classification or regression problem.

### 2.3.1. Binary classification

Each ground observation record denotes that if the date is before *DOY*, the phenological event has not taken place at the observation site; conversely, if the date is after *DOY*, the event has occurred. This definition enables the construction of multiple samples from each observation for binary classification using a sliding window and extraction of features from each window. If the window's end is before *DOY*, we assign label 0; otherwise, we assign label 1, indicating that the event has occurred. In the prediction phase, the first date at which the label changes from 0 to 1 can be considered to identify the predicted *DOY*. This concept is closely related to *survival analysis*, a statistical method used to examine and model the time until an event of interest occurs [13]. The utilization of a sliding window and a binary classifier has been recently employed in [12]. However, the authors recommended the direct use of time series as features for classification and did not implement a suitable and fair strategy for testing and model evaluation.

```

1 datasets:
2   npn_obs:
3     dataset: RNP
4     species_ids:
5       name: lilac
6       items: [36] # Syringa vulgaris / common lilac
7     phenophase_ids:
8       name: first leaf
9       items: [56,71,76,127,232,371,373,482,492,508]
10      years: [2000, 2021]
11      area:
12        name: US
13        bbox:
14          [
15            -171.791110603,
16            18.91619,
17            -66.96466,
18            71.3577635769
19          ]
20
21 daymet:
22   dataset: daymet_multiple_points
23   points:
24     source: npn_obs
25     years: [2000, 2021]
26     variables:
27       - tmax
28     resample:
29       frequency: dayofyear
30       operator: mean
31

```

**Fig. 2:** Springtime’s recipe (yaml file) for data preparation

### 2.3.2. Regression

ML-based phenological modeling can also be viewed as a regression problem. By considering the difference between the window’s end and the *DOY* as the target variable, this variable is negative when the event has not occurred and positive otherwise. Thus, in the prediction phase, the date at which the estimated target variable crosses zero can be recorded as the estimated *DOY*. It’s important to note that this approach differs significantly from considering *DOY* as the target variable in a regression model. In our perspective, regarding *DOY* as a target variable directly is not a practical approach for utilizing ML in predicting future unseen data.

## 3. EXPERIMENTAL RESULTS

Fig. 2 shows the *recipe* that we used in Springtime to prepare an integrated dataset. Using this *recipe*, we obtained 2010 observations. We exclude outliers by removing observations reporting leaf emergence before *DOY* equals 50 and after *DOY* equals 150. The data from 2021 is reserved for testing, while the remaining data is utilized for training and validation.

We opt for a window size of 30 days with a stride of 7 days, generating five samples before the *DOY* and five after. Regarding features, we extract the mean, median, minimum, and maximum temperature within each window. Additionally, we include counts of days within each window where the temperature exceeds 10, 15, and 20 degrees, as well as falls below 0, 5, and 10 degrees.

The XGBoost Classifier exhibited the best performance in binary classification. Employing this classifier, we achieved a mean absolute error (MAE) of 10.67 between the predicted *DOY* and the actual *DOY* in the test dataset. Fig. 3 (a) il-

lustrates the outcomes of using classification to predict the *DOY* of leaf emergence. Using the XGBoost Regressor, we obtained a slightly higher MAE, equal to 11.27. Fig. 3 (b) presents the results of using regression to predict the *DOY* of leaf emergence. Using the same training and test data, the TT process-based model obtained an MAE of 11.40, which is slightly higher than both ML-based approaches. Fig. 3 (c) depicts the results of phenological modeling using the TT process-based model.

## 4. CONCLUSION AND REMARKS

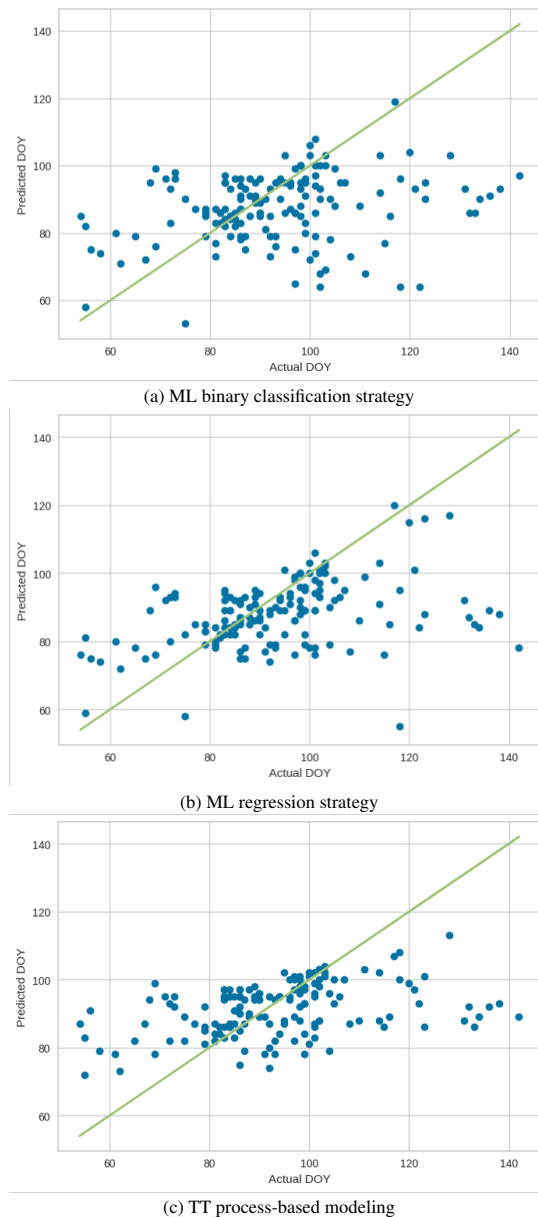
In this paper, we introduced a standardized workflow that leverages existing data, tools, and software packages. In particular, we proposed the use of the Springtime package, to facilitate the implementation and comparison of various phenology prediction models. We presented two approaches (classification and regression) for using ML methods in phenological modeling. Our experimental results highlight the efficacy of the XGBoost ML Classifier in predicting the *DOY* for leaf emergence in common lilac throughout the United States, achieving an MAE of 10.67, outperforming both the XGBoost Regressor and the traditional TT process-based model, which had MAEs of 11.27 and 11.40, respectively.

These findings demonstrate the potential of ML classification models to enhance phenological predictions. The workflow proposed in this paper aims to promote consistency and reliability in phenological modeling and model evaluation, addressing the current lack of standardization that hampers effective comparisons between different methodologies.

Future work should focus on refining ML models by incorporating additional environmental variables (e.g., remote sensing products) and extending the dataset to include a broader range of species and regions. Additionally, testing more recent process-based models needs to be considered. Another appealing future direction could be integrating ML and process-based approaches.

## 5. REFERENCES

- [1] Richardson AD, Keenan T, Migliavacca M, Ryu Y, Sonnentag O, and Toomey M, “Climate change, phenology, and phenological control of vegetation feedbacks to the climate system,” *AGRICULTURAL AND FOREST METEOROLOGY*, vol. 169, pp. 156–173, 2013.
- [2] Alyssa H. Rosemartin, Theresa M. Crimmins, Carolyn A.F. Enquist, Katharine L. Gerst, Jherime L. Kellermann, Erin E. Posthumus, Ellen G. Denny, Patricia Guertin, Lee Marsh, and Jake F. Weltzin, “Organizing phenological data resources to inform natural resource conservation,” *Biological Conservation*, vol. 173, pp. 90–97, 2014.



**Fig. 3:** Results of phenological modeling using different strategies.

- [3] Barbara Templ, Elisabeth Koch, Kjell Bolmgren, Markus Ungersböck, Anita Paul, Helfried Scheifinger, This Rutishauser, Montserrat Busto, Frank-M Chmielewski, Lenka Hájková, et al., “Pan european phenological database (pep725): a single point of access for european data,” *International journal of biometeorology*, vol. 62, pp. 1109–1113, 2018.
- [4] Brian J Stucky, Rob Guralnick, John Deck, Ellen G Denny, Kjell Bolmgren, and Ramona Walls, “The plant phenology ontology: A new informatics resource for large-scale integration of plant phenology data,” *Frontiers in Plant Science*, vol. 9, pp. 517, 2018.
- [5] Wanjing Li, Qinchuan Xin, Xuewen Zhou, Zhicheng Zhang, and Yongjian Ruan, “Comparisons of numerical phenology models and machine learning methods on predicting the spring onset of natural vegetation across the northern hemisphere,” *Ecological Indicators*, vol. 131, pp. 108126, 2021.
- [6] Shawn David Taylor, “Pyphenology: a python framework for plant phenology modelling,” *Journal of Open Source Software*, vol. 3, no. 28, pp. 827, 2018.
- [7] Koen Hufkens, David Basler, Tom Milliman, Eli K Melaas, and Andrew D Richardson, “An integrated phenology modelling framework in r,” *Methods in Ecology and Evolution*, vol. 9, no. 5, pp. 1276–1285, 2018.
- [8] Ahlem Hajjem, François Bellavance, and Denis Larocque, “Mixed-effects random forest for clustered data,” *Journal of Statistical Computation and Simulation*, vol. 84, no. 6, pp. 1313–1328, 2014.
- [9] Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana, “Interpretml: A unified framework for machine learning interpretability,” *arXiv preprint arXiv:1909.09223*, 2019.
- [10] Peter Kalverla, Fakhreh Alidoost, Stefan Verhoeven, and Mahdi Khodadadzadeh, “Springtime,” Apr. 2023.
- [11] Moez Ali, *PyCaret: An open source, low-code machine learning library in Python*, April 2020, PyCaret version 1.0.
- [12] Wujun Dai, Huiying Jin, Lin Zhou, Tong Liu, Yuhong Zhang, Zhiqiang Zhou, Yongshuo H Fu, and Guangze Jin, “Testing machine learning algorithms on a binary classification phenological model,” *Global Ecology and Biogeography*, vol. 32, no. 1, pp. 178–190, 2023.
- [13] Barbara Templ, Stefan Fleck, and Matthias Templ, “Change of plant phenophases explained by survival modeling,” *International Journal of Biometeorology*, vol. 61, pp. 881–889, 2017.