climpred: Verification of weather and climate forecasts

Riley X. Brady¹ and Aaron Spring², ³

¹ Department of Atmospheric and Oceanic Sciences and Institute of Arctic and Alpine Research, University of Colorado Boulder, Boulder, Colorado USA ² Max Planck Institute for Meteorology, Hamburg, Germany ³ International Max Planck Research School on Earth System Modelling, Hamburg, Germany

Summary

Predicting extreme events and variations in weather and climate provides crucial information for economic, social, and environmental decision-making (Merryfield et al., 2020). However, quantifying prediction skill for multi-dimensional geospatial model output is computationally expensive and a difficult coding challenge. The large datasets (order gigabytes to terabytes) require parallel and out-of-memory computing to be analyzed efficiently. Further, aligning the many forecast initializations with differing observational products is a straight-forward, but exhausting and error-prone exercise for researchers.

To simplify and standardize forecast verification across scales from hourly weather to decadal climate forecasts, we built climpred: a community-driven python package for computationally efficient and methodologically consistent verification of ensemble prediction models. The code base is maintained through open-source development. It leverages xarray (Hoyer & Hamman, 2017) to anticipate core prediction ensemble dimensions (ensemble member, initialization date and lead time) and dask (Dask Development Team, 2016; Rocklin, 2015) to perform out-of-memory and parallelized computations on large datasets.

climpred aims to offer a comprehensive set of analysis tools for assessing the quality of dynamical forecasts relative to verification products (e.g., observations, reanalysis products, control simulations). The package includes a suite of deterministic and probabilistic verification metrics that are constantly expanded by the community and are generally organized in our companion package, xskillscore.

Statement of Need

While other climate verification packages exist (e.g., s2dverification (Manubens et al., 2018) written in R and MurCSS (Illing et al., 2014) written with python-based CDO-bindings (Schulzweida, 2019)), climpred is unique for many reasons.

1. climpred spans broad temporal scales of prediction, supporting the weather, subseasonal-to-seasonal (S2S), and seasonal-to-decadal (S2D) communities.
2. climpred is highly modular and supports the research process from end-to-end, from loading in model output, to interactive pre-processing and analysis, to visualization.
3. climpred supports dask (Dask Development Team, 2016; Rocklin, 2015) and thus works across all computational scales, from personal laptops to supercomputers (HPC).
4. Flexibility and scaling leads to verification of global 5° x 5° resolution climate predictions in 8 seconds, compared to the 8 minutes required by MurCSS. However, note that climpred modularizes its workflow such that the verification step is performed on already pre-processed output, while MurCSS uses a more rigid framework that always required pre-processing. This time scale of seconds allows for a truly interactive analysis experience.

5. climpred is part of the wider scientific python community, pangeo (Abernathey et al., 2017; Eynard-Bontemps et al., 2019). A wide adoption of climpred could standardize prediction model evaluation and make verification reproducible (Irving, 2015).

Prediction Simulation Types

Weather and climate modeling institutions typically run so-called “hindcasts,” where dynamical models are retrospectively initialized from many past observed climate states (Meehl et al., 2009). Initializations are then slightly perturbed to generate an ensemble of forecasts that diverge solely due to their sensitive dependence on initial conditions (Lorenz, 1963). Hindcasts are evaluated by using some statistical metric to score their performance against historical observations. “Skill” is established by comparing these results to the performance of some “reference” forecast (Jolliffe & Stephenson (2012); e.g., a persistence forecast). The main assumption is that the skill established relative to the past will propagate to forecasts of the future.

A more idealized approach is the so-called “perfect-model” framework, which is ideal for investigating processes leading to potentially exploitable predictability (Bushuk et al., 2018; Griffies & Bryan, 1997; Séférian et al., 2018; Spring & Ilyina, 2020). Ensemble members are spun off an individual model (by slightly perturbing its state) to predict its own evolution. This avoids initialization shocks (Kröger et al., 2017), since the framework is self-contained. However, it cannot predict the real world. The perfect-model setup rather estimates the theoretical upper limit timescale after which the value of dynamical initialization is lost due to chaos in the Earth system, assuming that the model perfectly replicates the dynamics of the real world. Skill quantification is accomplished by considering one ensemble member as the verification data and the remaining members as the forecasts (Griffies & Bryan, 1997).

Climpred Classes and Object-Oriented Verification

climpred supports both prediction system formats, offering HindcastEnsemble and PerfectModelEnsemble objects. HindcastEnsemble is instantiated with an initialized hindcast ensemble dataset and requires an observational dataset against which to verify. PerfectModelEnsemble is instantiated with an initialized perfect-model ensemble dataset and also accepts a control dataset against which to evaluate forecasts. Both objects can also track an uninitialized dataset, which represents a historical simulation that evolves solely due to random internal climate variability or can be used to isolate the influence of external forcing (e.g., Kay et al., 2014).

Assessing skill for PredictionEnsemble objects (the parent class to HindcastEnsemble and PerfectModelEnsemble) is standardized into a one-liner:

```python
PredictionEnsemble.verify(
    # Score forecast using the Anomaly Correlation Coefficient.  
    metric='acc',
    # Compare the ensemble mean to observations.  
    comparison='e2o',
```

Each keyword argument allows flexibility from the user's end—one can select from a library of metrics, comparison types, alignment strategies, dimensional reductions, and reference forecasts. The most unique feature to climpred, however, is the ability for users to choose the alignment strategy to pair initialization dates with verification dates over numerous lead times. In other words, initialization dates need to be converted to target forecast dates by shifting them using the lead time coordinate. This is tedious, since one must remedy disparities in calendar types between the model and observations and account for the time span of or gaps in observations relative to the time span of the model.

There is seemingly no unified approach to how hindcast initialization dates are aligned with observational dates in the academic literature. The authors of climpred thus identified three techniques, which can be selected by the user:

1. Maximize the degrees of freedom by selecting all initialization dates that verify with the available observations at each lead. In turn, initializations and verification dates are not held constant for each lead.
2. Use the identical set of initializations that can verify over the given observational window at all leads. However, the verification dates change at each lead.
3. Use the identical verification window at each lead, while allowing the set of initializations used at each lead to change.

These strategies are shown graphically and explained in more detail in the documentation. Note that climpred offers extensive analysis functionality in addition to forecast verification, such as spatiotemporal smoothing (Goddard et al., 2013), bias removal (Boer et al., 2016), significance testing (Boer et al., 2016; DelSole & Tippett, 2016; Goddard et al., 2013), and a graphics library.

Use in Academic Literature

climpred has been used to drive analysis in three academic papers so far. Brady et al. (2020) used the HindcastEnsemble class to highlight multi-year predictability of ocean acidification in the California Current; Spring & Ilyina (2020) and Spring et al. (2021) used the PerfectModelEnsemble class to highlight predictability horizons in the global carbon cycle; and Krumhardt et al. (2020) used the HindcastEnsemble class to illuminate multi-year predictability in marine Net Primary Productivity.

Acknowledgements

We thank Andrew Huang for early stage refactoring and continued feedback on climpred. We also thank Kathy Pegion for pioneering the seasonal, monthly, and subseasonal time resolutions. Thanks in addition to Ray Bell for initiating and maintaining xskillscore, which serves to host the majority of metrics used in climpred.
References


predictions with the MPI earth system model: An initial shock in the North Atlantic. *Climate Dynamics*. https://doi.org/gdsnf8


