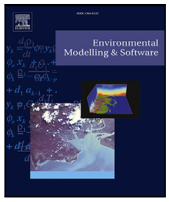
Environmental Modelling and Software 165 (2023) 105712

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft



CCdownscaling: A Python package for multivariable statistical climate model downscaling

Andrew D. Polasky [a], Jenni L. Evans [a,b,*], Jose D. Fuentes [a]

[a] Department of Meteorology and Atmospheric Science, The Pennsylvania State University, University Park, PA, USA
[b] Institute for Computational and Data Sciences, The Pennsylvania State University, University Park, PA, USA

ARTICLE INFO

Dataset link: https://zenodo.org/record/6506677, https://zenodo.org/record/7305359

Keywords:
Statistical downscaling
Self organizing maps
Climate change
Random forest
software
Python

ABSTRACT

Future climate projections are made with global numerical models whose spatial resolution often exceed 100s of km^2. These scales are too large to resolve many weather events, leaving a gap between the climate information needed to understand the impact of climate change on many human activities, and the information that can be provided by global models. Regional climate projections generated using statistical downscaling methods can provide an essential bridge between global climate models and the high spatial resolution data needed. As the demand for localized climate information continues to grow, new software tools are necessary to provide downscaled climate information. In this article, we describe CCdownscaling, a software package that provides multiple statistical climate downscaling methods to the station scale, including the Self Organizing Maps method. CCdownscaling includes several evaluation metrics for assessing the skill of downscaled climate information in various applications, and we demonstrate these features on an example dataset.

1. Introduction

General circulation models (GCMs) provide estimates for the state of the Earth's climate under a range of future greenhouse gas emissions scenarios. The GCMs typically run on horizontal spatial (grid) scales ranging from 0.7 × 0.7 degrees^2 to 2.5 × 2.5 degrees^2 (Taylor et al., 2012; Priestley et al., 2020). These spatial scales are too large to capture many weather events that are crucial for understanding the impacts of climate change on human populations (Radić and Clarke, 2011; Taylor et al., 2012). Therefore, it is often necessary to regionally generate downscaled climate information to the spatial scales required for the applications of interest (Maraun et al., 2010).

Precipitation is of particular interest and importance for projecting impacts of climate change on human activity. Unfortunately, precipitation remains a challenge for GCMs in many regions, for example the tropics, where the modeled precipitation can struggle to match even the annual mean precipitation (Koutroulis et al., 2016; Yang and Huang, 2022). As we move to higher spatial and temporal resolutions, the characteristics of precipitation become an even larger challenge for most GCMs to accurately reproduce, with GCM errors often being larger than size of projected changes (Zamani et al., 2020; Almazroui et al., 2021). Even in the regions where GCMs do capture the precipitation dynamics at the desired spatial scales, the amounts of precipitation represent an areal average on the model grid, thereby missing the characterization of localized extreme low or high precipitation events (Gervais et al., 2014). These characteristics create the so-called "drizzle" effect in GCMs, and result in projections with too many days of moderate precipitation and too few periods with no or extreme high precipitation (Stephens et al., 2010; Mehran et al., 2014; Koutroulis et al., 2016). Correcting these model biases remains a crucial endeavor for determining accurate estimates of the impacts of climate change, especially for accurately predicting changes to the frequency and severity of droughts and flooding (Camici et al., 2014; Quintero et al., 2018; Ahmadalipour et al., 2017).

Statistical downscaling of climate information is one option for reaching the desired spatial-scale resolution, and for bridging the gap between the information provided by the GCMs and that needed by users of climate information (Robinson and Finkelstein, 1991; Fowler et al., 2007). Statistical downscaling methods use empirical relationships between the outputs of GCMs and more localized (i.e., weather station) data to create quantitative models for scaling the given state of climate from the synoptic scale to local environments. This approach of incorporating GCM outputs and estimates of localized conditions can be used to produce future climate scenarios under different radiative forcing scenarios, such as the shared socioeconomic pathway (SSP) scenarios produced by the Intergovernmental Panel on Climate Change (Gidden et al., 2019).

* Correspondence to: Department of Meteorology and Atmospheric Science, The Pennsylvania State University, 503 Walker Building, University Park, PA 16801, USA.
E-mail address: jle7@psu.edu (J.L. Evans).

https://doi.org/10.1016/j.envsoft.2023.105712
Received 8 November 2022; Received in revised form 11 April 2023; Accepted 1 May 2023
Available online 4 May 2023

Numerous methods have been used for statistical downscaling, varying widely in complexity. Some of the earlier downscaling methods include bias-correction approaches to amend, scale, and produce outputs, based on the results from GCMs, that apply to the local conditions (Karl et al., 1990; Murphy, 1999). Subsequent approaches, based on a range of statistical and machine learning techniques, include the use of artificial neural networks (Hewitson and Crane, 1996; Ahmed et al., 2015; Hernanz et al., 2022), clustering (Hewitson and Crane, 2006; Wang et al., 2013), stochastic weather generators (Wilks, 1999; Kilsby et al., 2007), and constructed analogs (Abatzoglou and Brown, 2012; Pierce et al., 2014). These methods have improved the availability and reliability of downscaled climate projections, but the choice of downscaling method can have a significant impact on the results (Wang et al., 2017).

While downscaling is a widely used approach for estimating climate information at given locales (Maraun and Widmann, 2018), there is a dearth of easy-to-use software for producing and evaluating tailored downscaled climate projections. Some existing options include the Statistical Downscaling Model (SDSM, Wilby and Dawson, 2013), which uses a conditional weather generator approach, and the DownscaleR software package (Bedia et al., 2020) that provides several bias correction, linear regression, and analog methods. The SDSM provides only a single downscaling approach, thereby limiting its adaptability to diverse and broad applications. Other software packages, such as the R package "musica", exist and provide tools for validation of statistical downscaling methods at multiple time scales (Hanel et al., 2017).

As the need for regional climate information increases, there is demand to develop easy-to-implement numerical methods to generate regionally downscaled climate projections. One objective of this study is to describe the CCdownscaling (Climate Change Downscaling) software package that provides a framework for incorporating user-defined variables to generate climate projections at the station level. Notably, CCdownscaling includes the popular and theoretically sound Self-Organizing Map (SOM) downscaling method, which was not previously available in any publicly available software. An additional objective is to couple the provided downscaling methods with a framework for evaluating the skill of the generated climate projections. The package is written in Python because the language is widely used in both the Atmospheric Science and Machine Learning communities, making it an ideal choice for wide distribution and application. Given that applications of climate projections differ widely and apply to diverse environments, it is crucial to establish evaluation metrics for the downscaled products. In short, the goals of this paper are to:

- Describe an easy to use Python package implementing the SOM algorithm, as well as other downscaling methods.
- Describe a set of metrics for evaluating the reliability of downscaling methods included in the Python package.
- Demonstrate the use of the methods and metrics in the CCdownscaling package on an example downscaling use case.

1.1. Description of the CCdownscaling package

The CCdownscaling package is an open-source and freely available implementation of a number of downscaling approaches, designed for downscaling from GCM grid scale to station locations. The code has also been designed to easily accommodate new methods as desired by future users. CCdownscaling package provides a framework for using many common machine learning tools as downscaling methods with the goal to allow users to leverage existing and ongoing advances in machine learning when approaching downscaling problems. The package allows users to leverage the powerful scikit-learn (Pedregosa et al., 2011) and TensorFlow (Abadi et al., 2015) machine learning libraries to use common machine learning approaches for downscaling. In addition, the CCdownscaling package provides several point-based downscaling methods, as well as metrics for evaluating the skill of different methods on several variables important for different downscaling applications.

The SOM downscaling method (initially described in Hewitson and Crane (2006)) is not currently available in any publicly available software package. We have incorporated the SOM downscaling in CCdownscaling within a flexible framework. In doing so, we demonstrate how CCdownscaling can be easily extended to new machine learning methods, to allow for future additions to the downscaling package, and integrate with commonly used existing machine learning frameworks. This will allow for easy integration of future machine learning techniques. Using the scikit-learn package (Pedregosa et al., 2011), we provide a number of additional machine learning algorithms for comparison as downscaling methods, including random forest and multiple linear regression models.

The CCdownscaling package contains three primary components: pre-processing tools, downscaling methods, and evaluation metrics. The pre-processing tools include methods for variable selection and dimension reduction (Section 3), tools for selecting specific patterns of train and test data (Section 2.3). The downscaling methods (Section 4) provide a range of approaches for addressing the challenges of downscaling for different climate regimes and variables. And finally, the evaluation metrics (Section 5) provide tools for assessing the skill of the downscaling methods and inputs. An example showing the use of each of these components is provided in the form of a Python script and Jupyter notebook with the software package.

2. Downscaling for climate change with CCdownscaling

While GCMs are our best tools for overall assessment of future climate scenarios, there are many important impacts of climate change that are not well specified in GCMs. Extreme precipitation events, for example, are often underrepresented in GCMs, and are a major potential climate change impact (Fig. 1). GCM data often has significant biases compared to the observed values in the historical period. Correcting these biases is important before projecting to future scenarios. Downscaling can provide a method for bias correction, ensuring that the distribution of values matches the observed distribution. Statistical downscaling is important to provide a bridge between the best available GCMs and information needed to make critical decisions on costs and adaptations to climate change. These include providing inputs to downstream applications, including crop yield modeling and human comfort metrics (Charles et al., 2017; Dahl et al., 2019).

Downscaling of GCM simulations to the scales desired for these applications needs to replicate the conditions resulting from climate and provide information on variability due to the weather and local factors. In this section, we describe how the CCdownscaling package can be used for downscaling, describing both the methods for downscaling and the diagnostics included in CCdownscaling to evaluate the success of the downscaling based on key factors (e.g., statistical distribution, behavior of extremes, temporal correlation) important to the chosen situation.

2.1. Example case

Downscaling is generally useful to answer specific questions that cannot be adequately addressed by global models. In this paper, we will demonstrate the use of the CCdownscaling package to answer three such questions for a chosen location, using data from O'Hare airport near Chicago, Illinois. We will look at the frequency of days with a maximum temperature above the historical 90^{th} percentile, the change in the average rainfall, and change in frequency of precipitation above 20 mm per day.

To carry out this downscaling, we use precipitation and daily maximum temperature from the Global Summary of Day (GSOD, National Climatic Data Center, 2020) dataset for O'Hare Airport in Chicago,

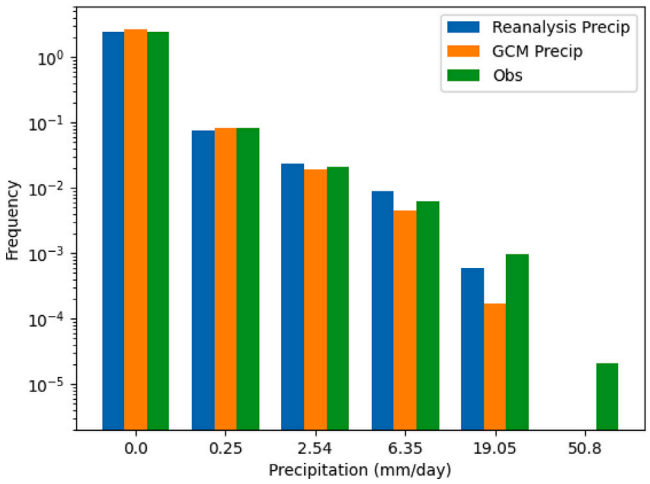


Fig. 1. Bar chart comparing the rainfall amounts for O'Hare Airport (Chicago, Illinois, USA) in the observed data, the NCEP reanalysis data, and an example GCM (the GFDL-ESM2M model from the Coupled Model Intercomparison Project Phase 5 (CMIP5)) for 1976–2005. The Reanalysis and GCM data underestimate the large rainfall events compared to the observations.

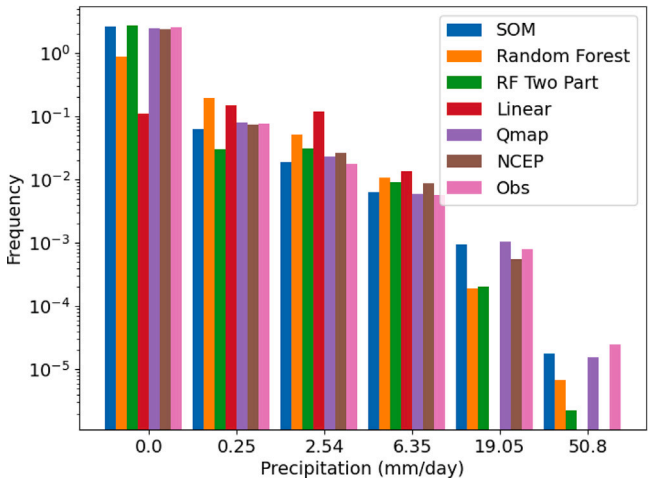


Fig. 2. Bar chart of precipitation amounts for O'Hare Airport for SOM and random forest downscaling, as well as the uncorrected NCEP precipitation values, and the station observations. The NCEP values underestimate particularly the large precipitation events. The two part random forest downscaling corrects this to some degree, but the SOM and Quantile Mapping (QMAP) methods do significantly better.

Illinois. Reanalysis data is taken from the National Center for Environmental Prediction (NCEP, Kalnay et al., 1996) reanalysis 2 for relative humidity (at 850 hPa), air temperature (850 hPa), geopotential height (500 hPa), sea level pressure (SLP, surface), and zonal and meridional wind components (700 hPa). These predictors were selected to capture the synoptic environment of the region. More information on the variables selection process can be found in Section 3.

This data is used to train the downscaling methods included in the CCdownscaling software package, and to evaluate the results using the various metrics described below. Results shown below for this example, and the required data sets and Python code are provided in the GitHub repository (https://github.com/drewpolasky/CCdownscaling). The period from 1976 to 1999 is used to train the downscaling methods, which are then tested on the years from 2000 to 2005. In the following sections we discuss the methods and evaluation metrics in the context of this example case.

The example use case is provided as both a Python script and Jupyter notebook in the package repository. Both provide the same code, with the Jupyter notebook providing a more interactive format. With the package installed, it can either be run directly in Python from the ohare_example.py file or the Jupyter notebook of the same name, located in the "example" folder. The data (in NetCDF format) needed to run the example can be downloaded from https://zenodo.org/record/7817799.

2.2. Reproducing climate variability

A downscaling method must be able to demonstrate that it can reproduce the variability of the observed climate from the input data. This is particularly important in cases (such as precipitation) where the variability is not well described in the GCMs (Stephens et al., 2010). The CCdownscaling package includes several metrics to evaluate the ability of a downscaling method to reproduce the existing climate variability, including Probability Density Function (PDF) skill score, Kolmogorov–Smirnov testing, and seasonal tests (see Section 5 for more details on these methods). These tests are carried out on an independent test set, to help separate from the data used to train the downscaling model. For our example case, we use the final 6 years of the 30 year period (2000–2005) as our test set. (Fig. 2).

2.3. Adapting to new climate conditions

A key challenge for statistical downscaling comes from trying to make predictions for future climate scenarios, where we are expecting the conditions to be significantly different from those we observe today. A good downscaling technique must therefore be able to adapt to changes in the underlying climate that go beyond the data it was trained on. Evaluating downscaling methods on this criteria requires some creativity, since the observational records typical cover smaller changes in climate than those we expect under most climate change scenarios (Gulev et al., 2021).

To address this difficulty, we implement two evaluation methods in the CCdownscaling package. For the first method, the input data is ranked over a given time period, and split such that the highest or lowest time periods are in the test set. For example, we select the six hottest years from 1976 to 2005 at O'Hare airport, and use those years for the test set, training on the other years. In this case, the six wettest years averaged 0.95 mm per day above the remaining years, an increase of 40%. To evaluate the different downscaling methods, we plot their performance on the biased train and test sets, to see how much of the change in precipitation they capture (Fig. 3). Selecting the train and test sets for this method can be done with the *select_max_target_years* function.

The second method takes advantage of the differences in climate between seasons to explore the ability of the methods to shift to new climates. We train a model on one season (e.g., spring), and evaluate that same model on another season (e.g., summer). In the O'Hare example, the summer is an average of 12.7 °C warmer than the spring, with 18.5% more precipitation, providing a more extreme test case than selecting the warmest set of years. This method requires tuning for individual locations when selecting the train and test season dates. For the O'Hare example, we use March–April–May (MAM) for spring, June–July–August (JJA) for summer. Other regions, such as tropical or monsoon environments, would require different date selections. This is customizable by setting the train and test dates in the *select_season_train_test* function.

These two functions for splitting train and test data are provided in addition to two more traditional techniques: a simple split, taking the most recent years for the test set, and a k-fold cross validation split, which forms a number of train/test split sets by dividing the available data into a given number (k) of segments. Each of these segments is

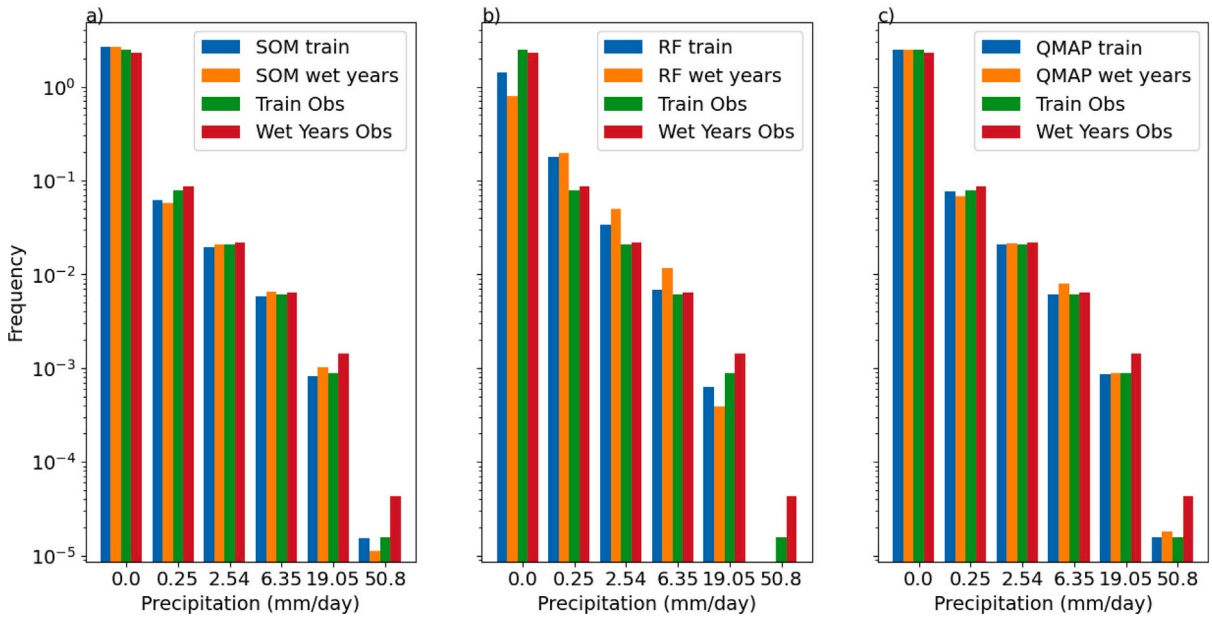


Fig. 3. Bar chart comparing the downscaling precipitation for the SOM (a), RF (b), and QMAP (c) methods when trained on the 24 years with the least precipitation from 1976–2005, and evaluated on the years with the most precipitation. None of the methods capture all of the shift to more precipitation, but the RF method does the best job, increasing in average precipitation by 0.59 mm per day, compared to 0.3 for the SOM and QMAP methods.

held out as the test data while the downscaling model is trained on the remaining data. This is especially useful in areas with limited input data, as a smaller test set size can be used while still maintaining a robust estimate of model skill (Fushiki, 2011).

Many of the changes projected under most climate change scenarios represent climates that have not been previously experienced, posing a challenge for training and verifying statistical downscaling models. Using these methods, we are able to test the ability of the downscaling methods to adapt to different circumstances to those they were trained on. In the example Jupyter notebook, examples for each of these splitting methods can be found in Section 2.

2.4. Considering extremes

One of the most critical pieces for understanding the impact of climate change is understanding changes in frequency and severity of extreme events (Katz and Brown, 1992). These events are also more difficult for GCMs to represent accurately than changes to the average of a given variable, increasing the need for downscaling methods that can capture such events (Kysel et al., 2002; Knutti and Sedláček, 2013).

In the CCdownscaling package, we consider two forms of extreme events: percentage-based and absolute. Percentage-based events are defined by the frequency of exceeding a given percentile of the observed data for a given location. For example, at O'Hare airport, the 90^{th} percentile for the training data of 1976–1999 is 30.6 °C, and we can calculate the number of days in the train and test sets that fall above this threshold.

Absolute metrics look at values that have specific meanings for impacts. For example, many crops suffer from decreased growth rates above certain temperature thresholds. Schlenker and Roberts (2009) found that average daily temperature of above 39 °C for corn, and 30 °C for soy caused yields to decline rapidly. CCdownscaling provides tools for assessing both of these types of metrics through the included Climdex module, which implements the 27 ETCCDI climate indices (Peterson, 2005). These indices cover a range of temperature and precipitation based metrics to provide an overview of the changes in key aspects of a region's climate. A full list of the indices can be found in Appendix A, and an example use of these metrics can be found in Section 9 of the example Jupyter notebook.

2.5. Comparing downscaling methods with CCdownscaling

The different downscaling methods provided by the CCdownscaling package have different strengths and weaknesses when answering different downscaling questions. Demonstrations of these differences can be found in the results in Section 5.

The SOM method (see Section 4.1 for more details on the method) is well suited to assessing changes in the frequency of events, and does a good job of recreating past climate variability (Ning et al., 2012). It can, however, struggle to respond to large changes in the underlying climate, and cannot extrapolate to new extreme values (Hewitson and Crane, 2006). It is better suited to analyses of how frequently an extreme event may occur in a future climate by estimating the frequency of synoptic conditions that have historically created extreme events, rather than estimating the maximum severity of events (Polasky et al., 2021).

The Random Forest method (Section 4.2) is highly adaptable, and generally does a better job of matching individual days than many of the other methods (He et al., 2016). However, it can also overfit to the data, and result in too many days with moderate values. This effect can be particularly acute for precipitation downscaling, with random forest models often producing to many days of moderate precipitation, and not enough dry days, or days of extreme rainfall (Fig. 3b).

Quantile mapping (Section 4.3) is generally very good at reproducing the historical climate variability, but often struggles to generalize to new climates, especially for more complex variables such as precipitation (Zhao et al., 2017). Quantile mapping is also vulnerable to variance inflation, where the variance in the downscaled output is higher than the variance in the observed data over a given area (Maraun, 2013; Cannon et al., 2015), because the variance at low resolutions tends to be lower than the variance at high resolutions. In mapping between the two distributions, the marginal difference in value is corrected, but the local variability in values is not included, leading to an overly strong spatial correlation between observed locations. Without further correction, this leads to spatial averages that overproduce extremes at either end of the distribution, especially for variables such as precipitation, that have lower spatial correlations.

Table 1
Variable importance rankings for maximum temperature for SIR, PCA, and RF methods. Table entries are NCEP variable names and pressure level in hPa. PCA is an unsupervised method, meaning there is no influence of the maximum temperature target variable on the rankings.

	SIR	PCA	RF
1	HGT 500	HGT 500	AIR surface
2	AIR surface	HGT 600	RHUM 850
3	HGT 600	HGT 700	RHUM surface
4	HGT 300	HGT 300	UWND surface
5	AIR 700	SLP surface	RHUM 700
6	SLP surface	VWND 600	VWND surface
7	HGT 850	UWND 600	AIR 850
8	VWND 600	AIR 600	RHUM 500
9	UWND 700	AIR 850	UWND 850
10	UWND 300	VWND 600	RHUM 600

3. Variable selection

Appropriate variable selection for downscaling is of critical importance for training a reliable and accurate downscaling model (Najafi et al., 2011; Hammami et al., 2012; Teegavarapu and Goly, 2018). There are a large number of variables that could potentially be useful when downscaling. For most downscaling targets, there are a large number of atmospheric variables that correlate in complex ways. In the case of precipitation, for example, useful variables might include relative humidity, sea level pressure, or other variables related to humidity, circulation patterns, or convective activity (Charles et al., 1999; Timbal et al., 2008; Maraun et al., 2010). The selection of variables for downscaling must also take into account the representation of those variables in the GCMs. Many potentially useful variables are not reliably simulated in GCMs, making them likely to be poor choices for a downscaling model (Cavazos and Hewitson, 2005; Teegavarapu and Goly, 2018).

In most cases, expert opinion is likely to be useful in selecting final sets of predictors that make physical sense for the climatology of the downscaling target region. It can then be useful to compare to the variables that objective methods identify as the most useful. These objective methods can also be used to select a subset of the initial predictor set, to improve model speed with minimal drops in performance.

CCdownscaling includes three methods for variable selection, Sliced Inverse Regression (SIR, Li, 1991), and Principal Component Analysis (PCA, Hotelling, 1933) and random forest (RF, Breiman, 2001). Different methods for variable selection can make different identifications of the most important variables, so it can be helpful to run multiple techniques, and look for common variables between the different approaches. CCdownscaling makes comparing these methods straightforward, all of these methods are implemented in the variable_selection code, and an example can be found in the example folder and Jupyter notebook. The relative importance rankings for an initial set of five predictors at five pressure levels from these three methods can be found in Table 1. The six input variables are air temperature (AIR), geopotential height (HGT), relative humidity (RHUM), sea level pressure (SLP, surface only), meridional wind speed (VWND), and zonal wind speed (UWND).

Two of the three methods, SIR and PCA, also serve as dimension reduction techniques. These method can be used to transform the high-dimension input data to lower dimension constructed features that maintain as much of the original information as possible (Ma and Zhu, 2013). This can be used to construct a set of new inputs that are a combination of the original inputs, and capture as much as possible of the variance of the downscaling target. These new variables can be used as inputs to the downscaling methods, reducing the computational requirements by having fewer input variables, while still retaining the original information.

4. Downscaling methods

The CCdownscaling package incorporates several downscaling methods, all conforming to a common framework for integration into downscaling workflows. The Application Programming Interface (API) mirrors the scikit-learn setup for machine learning models, which allows the easy integration of scikit-learn methods, and ease of use for those already familiar with scikit-learn. This approach allows for the development, testing, and comparison of different downscaling methods in a shared framework, as displayed in Fig. 4. The usage of the methods is demonstrated in Sections 4 and 5 of the example Jupyter notebook. All of the downscaling methods in the package are initially trained from reanalysis data. Once a model has been trained, it can then be used with GCM data to provide downscaled estimates for future climate scenarios.

4.1. SOM downscaling

Self-Organizing Maps (SOMs) are an unsupervised machine learning method for mapping a complex set of inputs onto a two-dimensional map of nodes, each representing a typical pattern observed in the input data (Kohonen, 1990). For downscaling, SOMs can be used to identify characteristic synoptic scale weather patterns, and relate those patterns to the observed local conditions (Hewitson and Crane, 2006). The SOM can then be used with GCM projections to explore changes to the frequency of these patterns in future climate scenarios (Gibson et al., 2016). SOMs have been broadly used for downscaling, particularly of precipitation, in regions such as South Africa (Hewitson and Crane, 2006), Florida (Sinha et al., 2018), and the Midwest United States (Polasky et al., 2021).

The SOM method begins by creating a set of nodes arranged in a two dimensional grid. Each of these nodes is defined by a vector, matching the size of one case of training data. To train the SOM, each element of the training data (in our example case, downscaling for O'Hare Airport, this data comes from the NCEP reanalysis, taking a 5 × 5 grid-point window around the station) is compared to the SOM nodes. The node whose vector is nearest (in Euclidean distance) to the training element is selected as the Best Match Unit (BMU),

$$BMU = min(W_v - I(t)) \tag{1}$$

where W_v is the weight vector for node v, I is the input dataset, and t is the index of the training element. The BMU vector is then incrementally updated towards the training element, as are the neighboring nodes to the BMU. Each node weight vector is updated

$$W_v(s+1) = W_v(s) + \Theta(u, v)\alpha \times (I(t) - W_v(s)) \tag{2}$$

where s is the current iteration of the training, α is the learning rate of the SOM, and Θ is the neighborhood function, governing how much the update effects the nodes near to the BMU. The value of Θ decreases exponentially the distance of the node to BMU. Adjusting the neighboring nodes in addition to the BMU has the effect of sorting similar nodes to be near to one another in the map. The overall update rate is governed by α. With each successive pass through the dataset, α is decreased to more rapidly converge to a stable map.

Once the SOM has been trained, each day in the training data can be placed on the map by finding the BMU. For each node of the SOM, this gives a set of days corresponding to that pattern. The station observations for those days can be combined to create a probability function of local values for each SOM node. To create new downscaled projections, GCM data can be mapped onto the SOM, matching each day to its BMU. The probability function of the downscaling target variable can then be sampled from, producing the downscaled value for that day. By iterating through the days included in the GCM data sets, we produce a daily downscaled time series for the given location. The SOMs were implemented in Python using the

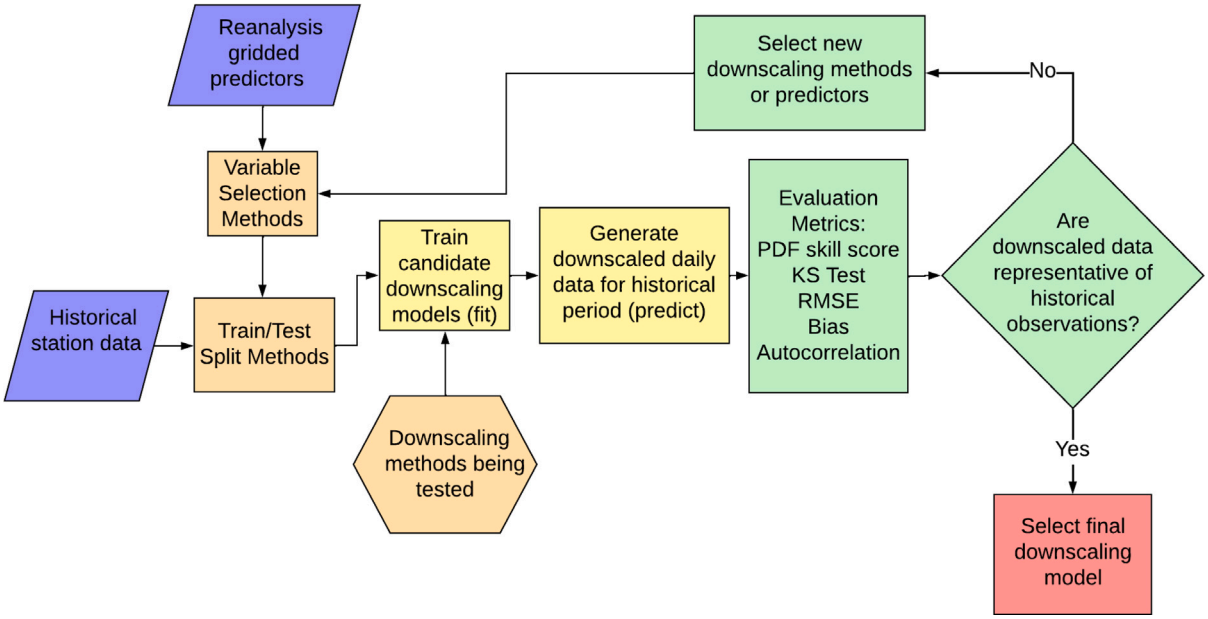


Fig. 4. This flowchart provides the sequence of training the downscaling process using historical data. Variables are selected from the reanalysis data for the target predictand, and the downscaling methods are trained and tested using the different evaluation metrics described in the text. In parenthesis are the names of the functions used for a given step in the downscaling process. The figure is adapted from Polasky et al. (2021). Once the downscaling model has been selected and trained, it can then be applied to future climate projections.

TensorFlow library (Abadi et al., 2015), adapted from the open-source TensorFlow-SOM project (Gorman, 2019).

The SOM method has the advantage of providing insight into the weather patterns giving rise to specific downscaled outcomes, through the patterns detected by the individual nodes of the SOM. In the O'Hare Airport example, the highest precipitation nodes fall in the center of the top two rows of the SOM (Fig. 5). The color gradient represents the frequency of that nodes being the BMU in the training data, while the number in the box is the average precipitation for the days falling on that node. Two of these nodes for example, the node at row 4, column 5 (4,1), and node (3,4), both have high precipitation, but very different temperature patterns (Fig. 6). Node (3,4) represents a warm summer day, with temperatures between 17 °C in the south of the region around O'Hare, and 11 °C in the North at 850 hPa. The high precipitation likely corresponds to summer-type convection. The (4,1) node is colder, with a strong southwest/northeast temperature gradient, and the precipitation is likely driven by mid-latitude cyclones.

Fig. 5. Heat map of the SOM nodes for O'Hare Airport. The color represents the number of observations of occurrence of that node, while the number in each box is the average precipitation on the days in that node cluster.

4.2. Scikit-learn downscaling methods

Scikit-Learn is a widely used machine learning library for Python that provides a wide range of machine learning tools (Pedregosa et al., 2011). These methods have been adapted for use as downscaling tools in CCdownscaling, as demonstrated for a random forest model for the O'Hare Airport example. Random forests are a widely used machine learning approach, that have been successfully applied to a wide range of problems (Breiman, 2001). Their adaptability and ease of use have led to random forest being a go-to method in machine learning (Biau and Scornet, 2016). Random forests have been used for downscaling temperature and precipitation in a variety of locations (Hutengs and Vohland, 2016; Sa'adi et al., 2017; Pang et al., 2017; Polasky et al., 2021).

The CCdownscaling package provides a framework for extending scikit-learn provided methods to better suit downscaling problems. An example of this functionality is given in two_step_random_forest model, which adapts the standard random forest from scikit-learn to have a classifier to initially split dry/precipitation days, then a second regressor model to predict the amount of precipitation for the wet days. This model addresses issues of the random forest producing too many days of middling precipitation, and too few dry days, similar to the undownscaled GCMs, and generally outperforms a basic random forest for precipitation metrics (Section 5).

4.3. Quantile mapping

Quantile mapping is a commonly used approach for downscaling, especially for temperature (Maraun, 2013; Han et al., 2019). Unlike the other methods included in this software package, quantile mapping uses only a single variable as input, typically the value being downscaled (i.e., daily precipitation from the model for downscaling precipitation). It takes the approach of comparing the difference in value between the quantile ranks of an initial and final distribution.

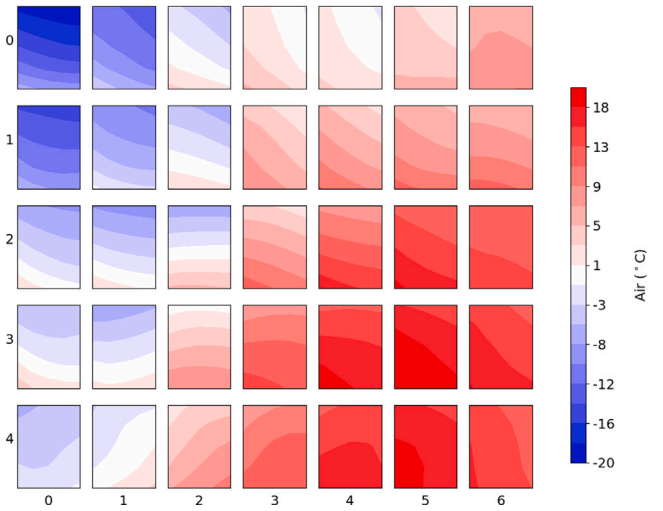


Fig. 6. Average daily 850 hPa air temperature (°C) for each node in the 5 × 7 SOM for the O'Hare Airport example. Each node is made up of a 5 × 5 slice of model grid points centered around the station. The strongest distinction is between primarily summer days in the lower right, and cold winter days in the upper left, with more intermediate patterns in between. There is also a separation on negatively/positively tilted temperature gradients moving from top to bottom.

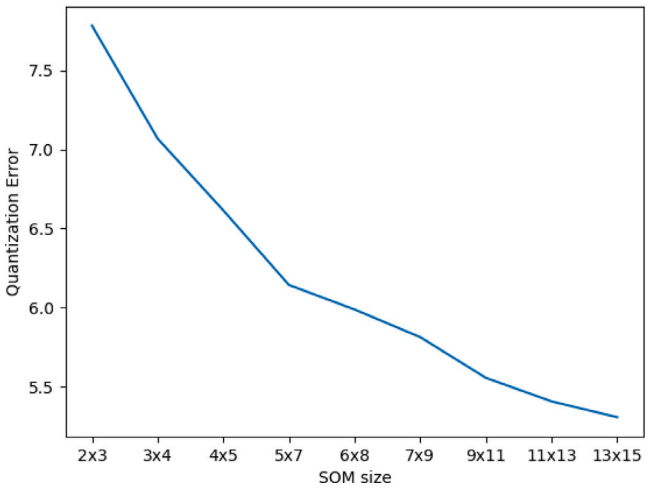


Fig. 7. Quantization error for SOM grids of increasing size. Quantization error decreases as the number of nodes increases, as the specificity of each node increases. Choosing the correct size of SOM grid is a trade off between the increasing complexity of the graph, and thus potential for overfitting, and the decreasing quantization error with larger maps. For this example, a grid size of 5 × 7 was chosen, as representing the "elbow" where rate of decrease in the QE slows down.

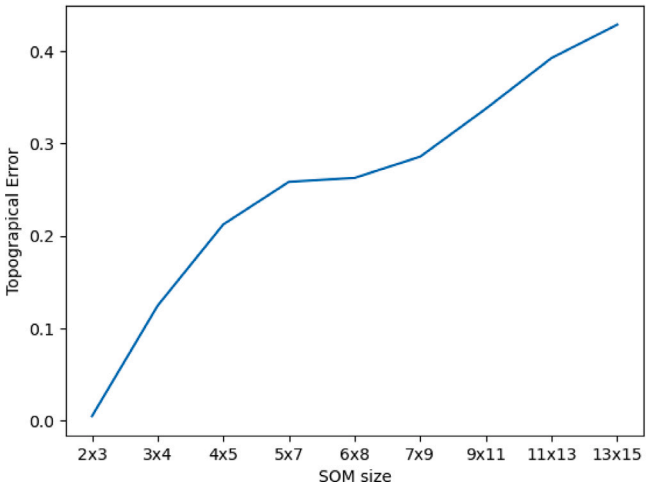


Fig. 8. Topographic error for increasing SOM grids sizes for the same example as Fig. 7. TE increases monotonically with the size of the grid.

A transformation is calculated between these ranks, which can then be applied to a new input distribution, to create a downscaled output that better matches the observed distribution of events.

Quantile mapping is effective at bias correcting GCMs outputs, but quantile mapping relies on the fields of the target variable from the GCMs. This can be an issue, particularly in the case of precipitation, where the representation of the processes underlying the output values are not properly resolved by the GCMs. Other methods, which can make use of variables that are better captured in the GCMs, are likely to yield superior results of the downscaling in these circumstances (Zhao et al., 2017). Nonetheless, quantile mapping provides a straightforward bias correction approach, and is a useful point of comparison for other downscaling methods (Wood et al., 2004).

5. Validation methods

In addition to providing a range of downscaling methods, CCdownscaling also provides a number of different evaluation metrics to compare methods and assess the suitability of a downscaling output for a given task. Depending on the goals of the downscaling and even the use case, different downscaling applications may best be assessed using different evaluation criteria. For example, estimating drought conditions will require accurate estimation of average precipitation and temperature over longer time frames, while projecting for flooding situations will require accurate representation of large precipitation events, and temporal correlation for multi-day events. The implementation of many validation methods in CCdownscaling is thus a key part of its broad applicability. In this section, we describe a number of evaluation metrics and show the results of these metrics on the O'Hare Airport example for a range of downscaling methods. Examples of these metrics can be found in Sections 7 and 8 of the example Jupyter notebook.

5.1. SOM training metrics

The SOM method includes two specialized training metrics: quantization and topological error. These are commonly used metrics for assessing the training characteristics of a model setup, and should be used when tuning the hyperparameters (controllable parameters that define the setup and training characteristics) of the model.

Quantization error (QE) refers to the average distance between the day vectors assigned to a node, and the characteristic vector of the node (Kohonen, 1990). Smaller values of QE represent reduced spread within each cluster. As the size of the map increases, this value will naturally decrease, as days are increasingly subdivided between clusters. A common tactic for settling on a map size is to look for the "elbow" where the decrease in QE slows down, with the increase in map size (Céréghino and Park, 2009). In Fig. 7, this occurs at the 5 × 7 map size, and that map size was selected for further analyses.

Topographical Error (TE) is calculated as the percentage of input vectors whose second-best matching units are adjacent to their best match units. This is a measure of how well the topology of the original dataset is being preserved in the lower-dimension space of the SOM (Uriarte and Martín, 2005). TE generally increases as the size of the map increases, with more opportunities for non-adjacent nodes to match an input vector (Fig. 8). These two metrics combine to help choose an optimal size for the SOM, balancing the gains in specificity against the increasing complexity of the model. An example demonstrating both of these methods can be found in *som_training_example.py* in the example folder of the package.

Table 2
Results scores for several different downscaling methods using the O'Hare airport example precipitation data. There are noticeable difference between the SOM and random forest (RF) scores, with the SOM showing a much higher RMSE than the RF but a higher PDF skill score, indicating that while the SOM does not match the individual days as well as the random forest model, it provides a better match of the distribution of events. The NCEP data come from the NCEP reanalysis precipitation for the grid point nearest to O'Hare Airport, without any downscaling applied.

	PDF skill score	KS test statistic	RMSE (mm/day)	Bias (mm/day)
SOM	0.99	0.03	11.03	0.04
RF	0.97	0.64	8.57	−0.48
RF two part	0.97	0.08	8.51	0.45
Qmap	0.99	0.17	9.97	−0.18
NCEP	0.97	0.65	8.88	0.12

5.2. Distribution tests

One of the key characteristics for a downscaling method is determining the skill of a downscaling approach is how well it reproduces the historical distribution of events. Precipitation, in particular, is commonly spread over too many days with small amounts of precipitation in GCMs projections, reducing the number of dry days and large precipitation events (Fig. 2). These metrics are especially valuable when trying to understand the shifts in climate, and the frequency of different event types under climate change scenarios (Perkins et al., 2013; Polasky et al., 2021). We implement two common distribution tests in CCdownscaling: PDF skill score and the Kolmogorov–Smirnov test.

The PDF skill score measures the similarity between two probability density functions (PDFs), by calculating the minimum value between observed and modeled counts within each bin of a histogram to measure the shared area within the distribution:

$$S_{score} = \sum_{1}^{n} min(Z_m, Z_o) \tag{3}$$

where S_{score} is the skill score, Z_m and Z_o are the frequency of modeled and observed values in a given bin, respectively, and n is the number of bins used to calculate the PDF (Perkins et al., 2007). PDF skill score provides an easily-interpretable score for the similarity between the observed and downscaled distribution of values.

The Kolmogorov–Smirnov (KS) test provides a non-parametric statistical test to evaluate the likelihood that two samples are drawn from the same underlying distribution (Massey, 1951). The KS test provides both a test statistic, which can be used to assess the similarity of two distributions (where smaller values are more similar), and a probability that the given distributions are drawn from the same underlying probability distribution. Tables 2 and 3 show the KS scores and probabilities for the different methods for the O'Hare Airport example for precipitation and maximum air temperature, respectively.

In most cases, PDF score and KS test statistic are strongly correlated (Brands et al., 2012). An exception occurs when the distribution includes a large number of near zero values, as in the case of daily precipitation. In this instance, PDF score can struggle due to the kernel density smoothing applied in the calculation (Brands et al., 2012). The effect of this difference can be seen in the O'Hare Airport example. For temperature, the SOM and random forest methods score similarly to one another in both PDF score and KS statistic (Table 3). By contrast, for precipitation, the PDF statistic for the two methods is similar, while the SOM far outperforms the random forest on KS statistic. The two-part random forest achieves a KS statistic score much closer to that of the SOM, primarily by correcting the number of days with no precipitation compared to the single random forest. Visually, we can see that the SOM much better matches the observed distribution (Fig. 2), indicating that the KS test statistic better represents the relative skill of the two downscaling methods for precipitation.

Table 3
As in Table 3, but for daily maximum temperature. The SOM and Quantile mapping (Qmap) methods have somewhat higher PDF skill scores, while the Qmap method does the best job of matching the day-to-day values, with the lowest RMSE.

	PDF skill score	KS test statistic	RMSE (K)	Bias (K)
SOM	0.96	0.02	8.00	0.26
RF	0.94	0.04	3.24	−0.36
Qmap	0.96	0.03	2.76	−0.25

Table 4
Climdex indices for maximum temperature, using the 6 warmest years as the test set. See Appendix A for index definitions.

	OBS	SOM	RF	QMAP
TxMean	17.37	16.31	16.63	16.54
TxMin	−13.59	−18.21	−11.41	−14.41
TxMax	37.77	36.70	37.60	37.35
Su25	127.5	112.7	112.7	117.7
ID0	31.33	41.17	31.17	34.83
Tx90p	60.17	35.17	39.83	43.67
Tx10p	31.33	41.17	31.17	34.83
WSDI	29.33	0	11.67	22.33

Table 5
Climdex indices for precipitation, using the 6 wettest years as the test set. Here, the poor performance of the RF method on the high end of precipitation is very evident, while the SOM and QMAP methods generally perform similarly. See Appendix A for index definitions.

	OBS	SOM	RF	QMAP
PrcpMean	3.35	2.56	3.15	2.69
Rx1Day	93.7	56.8	38.6	68.1
Rx5Day	121	83.6	75.0	87.8
R95p	54.8	39.8	18.2	38.8
R99p	87.4	59.4	27.7	64.3
SDII	8.06	7.25	3.55	6.61
CDD	12.7	12.8	1.50	12.2
CWD	2.17	1.33	1.33	1.50
R10mm	36.0	29.0	29.0	30.7
R20mm	18.5	12.3	3.83	10.7

5.3. Error metrics

Error metrics are a common method for evaluating model skill, evaluating the day by day differences in downscaled and observed values for the testing period. These direct measures of difference are most useful when comparing methods that seek to recreate the specific conditions on each individual day, such as the random forest model. Much of the goal with downscaling, and with any climate modeling, is to understand the full range and frequency of events occurring, rather than to predict the correct event on the specific day. For this reason, the error metrics are often less useful for assessing downscaling model skill than the distribution tests. Nonetheless, these measures can provide useful comparisons, especially between similar classes of models, and so we have made several error metrics available in CCdownscaling.

These error metrics should only be applied when comparing downscaled outputs calculated from reanalysis data to the ground-truth observations. Data sets coming from GCMs are unlikely to match the daily variation of the historical records, and the results may not represent the documented conditions. Commonly used error metrics such as mean squared error and bias are included in the package. The results for the O'Hare Airport example can be found in Table 2 for daily precipitation, and Table 3 for maximum temperature. The results for SOM and linear regression models for temperature show the differences between the error and distribution metrics. The SOM outperforms the linear model on the PDF and KS scores, as it provides a better performance of matching the overall distribution of events, but scores worse in terms of RMSE and bias, because it does not match the specific day-to-day variations as well as the linear model.

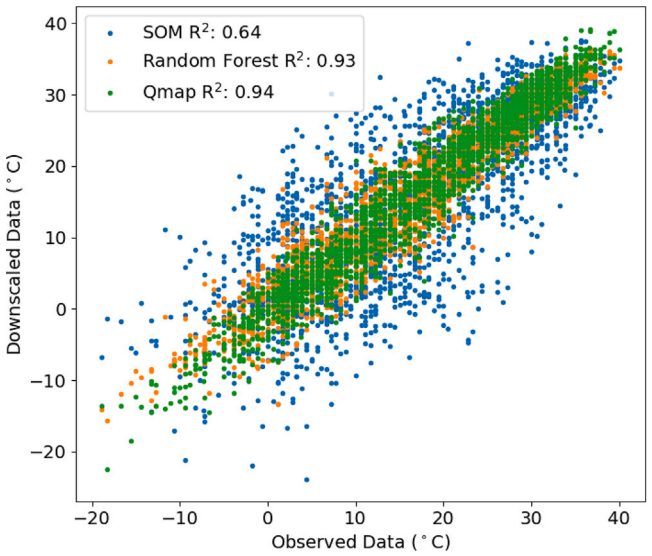


Fig. 9. Scatter plot of the reanalysis downscaled (Y axis) against the observed maximum temperature for the O'Hare station. The R^2 value for a linear regression between the observed and downscaled data is shown in the legend.

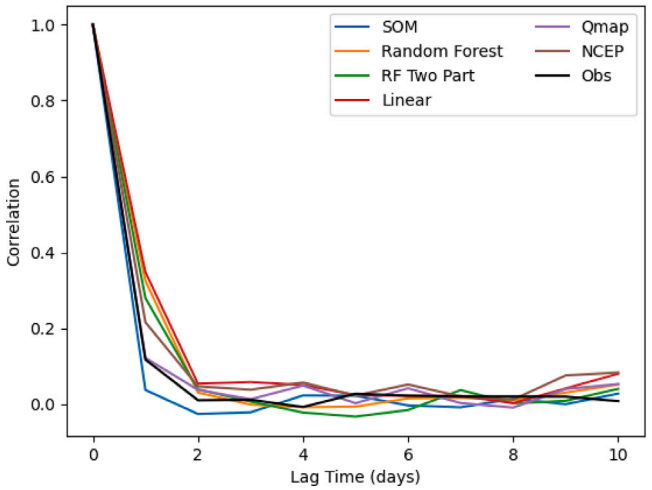


Fig. 10. Auto correlation for the different downscaling methods for precipitation at O'Hare airport.

A similar approach can be taken by plotting the observed and downscaled data against one another, and calculating a linear regression (Fig. 9). This provides a clear visual representation of how well the downscaled data matches the observations, but suffers from the same drawbacks as the error metrics mentioned above.

5.4. Extreme events

Performance on extreme events is a key goal for many downscaling applications (Wigley, 1985). Many extreme event definitions will depend on the specifics of the question to be answered, but we include some common extreme event indices in the CCdownscaling package. In addition to being potentially useful values for questions about future climate scenarios, these indices can be a useful metric for determining the skill of the different downscaling methods. Using the warmest (wettest) year split described in Section 2.4, we calculate the maximum temperature (precipitation) climdex values for the SOM, RF, and QMAP methods (Tables 4 and 5). For precipitation, while the RF method does the best job of the three at adapting the to higher average precipitation, the poor performance on the high and low ends of the distribution are evident in the very low values for 95^{th} percentile (R95p), 99^{th} percentile (R99p), and consecutive dry days (CDD). The differences are not as pronounced for the max temperature indices, but here QMAP clearly outperforms the other two methods for most of the indices.

5.5. Autocorrelation

Autocorrelation provides a metric for temporal similarity within time series data. It is calculated by taking the correlation between a timeseries and a time-lagged copy of the same timeseries, for a number of different time lags. This can be particularly important for applications such as flooding, where understanding the likelihood of multi-day large rainfall events is crucial for projecting the frequency of major flooding events in a given climate scenario. In the case of O'Hare Airport, the autocorrelation (of both the observed and downscaled datasets) falls off rapidly, with most of the downscaled methods, apart from the SOM and Qmap, overestimating the observed correlation at a lag of one day (Fig. 10).

6. Concluding remarks

The accelerating need for reliable, highly localized data for climate change scenarios has led to the development of software packages such as CCdownscaling that can readily and reliably provide information tailored to individual sites or regions. CCdownscaling is a new software package providing options for downscaling approaches and tools in a manner that the user can readily apply and iterate upon to meet the needs and requirements of the desired downscaled climate change projections. It is freely available and open-source, to be easily used in the creation of new downscaled climate data for a range of potential uses. CCdownscaling provides SOM-based downscaling, along with the extension of traditional machine learning tools to downscaling. We demonstrate the capabilities of the package using an example station located at the Chicago O'Hare Airport in Illinois. This example serves to demonstrate the use of the software package, but will not cover many issues that will come up for a real downscaling use case, including limited data availability, complex terrain, or changes in land cover.

Typically, we have found the SOM method performs well for precipitation downscaling, while the random forest and QMAP methods often perform well for temperature variables, but these results can vary based on the meteorology and available data for a given location. In addition to the downscaling methods, the package includes a variety of metrics for assessing the skill and utility of downscaled outputs. These tools, along with the example given, provide a framework for creating and evaluating new downscaled products, tailored to a given location and use case.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Jenni L. Evans reports financial support was provided by National Science Foundation.

Data availability

The data used in this article are available at: https://zenodo.org/record/6506677 (data) and https://zenodo.org/record/7305359 (code)

Acknowledgments

Thanks to Zach Moon for assistance in making the code more usable, and Daniel Polasky for helpful comments on the manuscript. We also thank two anonymous reviewers for their helpful comments on this manuscript. Thanks as well to the Penn State Institute for Computational and Data Sciences for computational resources and support. This work was supported by the National Science Foundation, United States under grant 1639342.

Table A.1
ETCCDI climdex indices and definitions.

Index	Definition
TxMean	Average Daily Maximum Temperature
TxMin	Average Minimum Daily Maximum temperature per year
TxMax	Average Maximum daily maximum temperature per year
Su25	Number of "Summer Days" above 25 C
ID0	Number of "Icing Days" where maximum temperature is below 0 C
Tx90p	Days per year above the 90^{th} percentile of maximum temperature in the training data
Tx10p	Days per year below the 10^{th} percentile of maximum temperature in the training data
WSDI	Warm Spell Duration Index: The number of days with at least 6 consecutive days above the reference 90^{th} percentile
TnMean	Average daily minimum temperature
TnMin	Average minimum daily minimum temperature per year
TnMax	Average maximum daily minimum temperature per year
Tn90p	Days per year above the 90^{th} percentile of minimum temperature in the training data
Tn10p	Days per year below the 10^{th} percentile of minimum temperature in the training data
CSDI	Cold Spell Duration Index: The number of days with at least 6 consecutive days below the reference 10^{th} percentile
FD	Frost Days, number of days with minimum temperature below 0 C
TR	Tropical Nights, number of days with minimum temperature above 20 C
PrcpMean	Average precipitation per day
Rx1Day	Maximum 1-day precipitation
Rx5Day	Maximum consecutive 5-day precipitation
R95p	Annual Total Precipitation when daily rainfall is above the 95^{th} percentile
R99p	Annual Total Precipitation when daily rainfall is above the 99^{th} percentile
SDII	Simple Precipitation Intensity Index
CDD	Consecutive Dry Days, maximum length per year
CWD	Consecutive Wet Days, maximum length per year
R10mm	Days per year with precipitation above 10 mm
R20mm	Days per year with precipitation above 20 mm

Appendix A. Climdex indices

The 27 Climdex Indices represent a standard set of values for precipitation, maximum temperature, and minimum temperatures to describe changes in climate, with a focus on extreme events, created by the Expert Team on Climate Change Detection and Indices (ETCCDI, http://etccdi.pacificclimate.org/indices.shtml). These values provide an overview of mean and extreme changes that will be commonly useful to many applications for maximum and minimum temperature, as well as precipitation (see Table A.1).

Appendix B. Package setup

Installation instructions for the CCdownscaling package can be found on the github, along with the code for the example use case, in the form of both a Python script and Jupyter notebook. Both Jupyter notebook and script contain the same code, the Jupyter notebook is included to provide a more readable and interactive interface for the example. To set up the environment for CCdownscaling, we recommend using Conda for ease of installation. A Conda *environment.yml* file with the list of requirements.

All the downscaling models in CCdownscaling conform to a common API, making the addition of new methods to a downscaling use case straightforward. The Jupyter notebook in the example folder gives an example of the main pieces of the CCdownscaling package. Section 2 shows an example for using three different train/test splits. Section 5 demonstrates training several different models, all called using the same format, where the model is created, then trained (fit), and final downscaling output is generated (predict):

```
som = som_downscale.som_downscale(som_x = 7, som_y = 5, batch = 512,
alpha = 0.1, epochs = 50)
rf_two_part = correction_downscale_methods.two_step_random_forest()
random_forest = sklearn.ensemble.RandomForestRegressor()
qmap = correction_downscale_methods.quantile_mapping()

#train
som.fit(training_data, train_hist, seed = 1)
random_forest.fit(training_data, train_hist)
rf_two_part.fit(training_data, train_hist)
qmap.fit(rean_precip_train, train_hist)

#generate outputs from the test data
som_output = som.predict(test_data)
random_forest_output = random_forest.predict(test_data)
rf_two_part_output = rf_two_part.predict(test_data)
qmap_output = qmap.predict(rean_precip_test)
```

The outputs from these different methods can then be passed into any of the analysis methods. Section 6 demonstrates some of the SOM specific analysis plots, including heatmaps and variable grids for the SOM nodes. Section 7 then shows the more universal evaluation metrics, computing skill score values for each of the methods. Section 8 makes the plots for comparing the different methods, and finally Section 9 calculates the climdex values for each output.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL https://www.tensorflow.org/. Software available from tensorflow.org.

Abatzoglou, J.T., Brown, T.J., 2012. A comparison of statistical downscaling methods suited for wildfire applications. Int. J. Climatol. 32 (5), 772–780.

Ahmadalipour, A., Moradkhani, H., Svoboda, M., 2017. Centennial drought outlook over the CONUS using NASA-NEX downscaled climate ensemble. Int. J. Climatol. 37 (5), 2477–2491.

Ahmed, K., Shahid, S., Haroon, S.B., Xiao-Jun, W., 2015. Multilayer perceptron neural network for downscaling rainfall in arid region: A case study of Baluchistan, Pakistan. J. Earth Syst. Sci. 124 (6), 1325–1341.

Almazroui, M., Islam, M.N., Saeed, F., Saeed, S., Ismail, M., Ehsan, M.A., Diallo, I., O'Brien, E., Ashfaq, M., Martínez-Castro, D., et al., 2021. Projected changes in temperature and precipitation over the united states, central America, and the caribbean in CMIP6 GCMs. Earth Syst. Environ. 5, 1–24.

Bedia, J., Baño-Medina, J., Legasa, M.N., Iturbide, M., Manzanas, R., Herrera, S., Casanueva, A., San-Martín, D., Cofiño, A.S., Gutiérrez, J.M., 2020. Statistical downscaling with the downscaler package (v3. 1.0): Contribution to the VALUE intercomparison experiment. Geosci. Model Dev. 13 (3), 1711–1735.

Biau, G., Scornet, E., 2016. A random forest guided tour. Test 25 (2), 197–227.

Brands, S., Gutiérrez, J.M., Herrera, S., Cofiño, A., 2012. On the use of reanalysis data for downscaling. J. Clim. 25 (7), 2517–2526.

Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Camici, S., Brocca, L., Melone, F., Moramarco, T., 2014. Impact of climate change on flood frequency using different climate models and downscaling approaches. J. Hydrol. Eng. 19 (8), 04014002.

Cannon, A.J., Sobie, S.R., Murdock, T.Q., 2015. Bias correction of GCM precipitation by quantile mapping: how well do methods preserve changes in quantiles and extremes? J. Clim. 28 (17), 6938–6959.

Cavazos, T., Hewitson, B.C., 2005. Performance of NCEP-NCAR reanalysis variables in statistical downscaling of daily precipitation. Clim. Res. 28 (2), 95–107.

Céréghino, R., Park, Y.-S., 2009. Review of the self-organizing map (SOM) approach in water resources: Commentary. Environ. Model. Softw. 24 (8), 945–947.

Charles, S.P., Bates, B.C., Whetton, P.H., Hughes, J.P., 1999. Validation of downscaling models for changed climate conditions: Case study of southwestern Australia. Clim. Res. 12 (1), 1–14.

Charles, B.C., Elijah, P., Vernon, R.C., 2017. Climate change impact on maize (Zea mays L.) yield using crop simulation and statistical downscaling models: A review. Sci. Res. Essays 12 (18), 167–187.

Dahl, K., Licker, R., Abatzoglou, J.T., Declet-Barreto, J., 2019. Increased frequency of and population exposure to extreme heat index days in the United States during the 21st century. Environ. Res. Commun. 1 (7), 075002.

Fowler, H.J., Blenkinsop, S., Tebaldi, C., 2007. Linking climate change modelling to impacts studies: Recent advances in downscaling techniques for hydrological modelling. Int. J. Climatol.: A J. R. Meteorol. Soc. 27 (12), 1547–1578.

Fushiki, T., 2011. Estimation of prediction error by using K-fold cross-validation. Stat. Comput. 21, 137–146.

Gervais, M., Tremblay, L.B., Gyakum, J.R., Atallah, E., 2014. Representing extremes in a daily gridded precipitation analysis over the United States: Impacts of station density, resolution, and gridding methods. J. Clim. 27 (14), 5201–5218.

Gibson, P.B., Perkins-Kirkpatrick, S.E., Renwick, J.A., 2016. Projected changes in synoptic weather patterns over New Zealand examined through self-organizing maps. Int. J. Climatol. 36 (12), 3934–3948.

Gidden, M.J., Riahi, K., Smith, S.J., Fujimori, S., Luderer, G., Kriegler, E., Van Vuuren, D.P., Van Den Berg, M., Feng, L., Klein, D., et al., 2019. Global emissions pathways under different socioeconomic scenarios for use in CMIP6: A dataset of harmonized emissions trajectories through the end of the century. Geosci. Model Dev. 12 (4), 1443–1475.

Gorman, C., 2019. Tensorflow-som. https://github.com/cgorman/tensorflow-som.

Gulev, S., Thorne, P., Ahn, J., Dentener, F., Domingues, C., Gerland, S., Gong, D., Kaufman, D., Nnamchi, H., Quaas, J., Rivera, J., Sathyendranath, S., Smith, S., Trewin, B., von Schuckmann, K., Vose, R., 2021. In: Masson-Delmotte, V., Zhai, P., Pirani, A., Connors, S., Péan, C., Berger, S., Caud, N., Chen, Y., Goldfarb, L., Gomis, M., Huang, M., Leitzell, K., Lonnoy, E., Matthews, J., Maycock, T., Waterfield, T., ci, O.Y., Yu, R., Zhou, B. (Eds.), 2021: Changing State of the Climate System. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, pp. 287–422. http://dx.doi.org/10.1017/9781009157896.004.

Hammami, D., Lee, T.S., Ouarda, T.B., Lee, J., 2012. Predictor selection for downscaling GCM data with LASSO. J. Geophys. Res.: Atmos. 117 (D17).

Han, Z., Shi, Y., Wu, J., Xu, Y., Zhou, B., 2019. Combined dynamical and statistical downscaling for high-resolution projections of multiple climate variables in the Beijing–Tianjin–Hebei region of China. J. Appl. Meteorol. Climatol. 58 (11), 2387–2403.

Hanel, M., Kožín, R., Heřmanovský, M., Roub, R., 2017. An R package for assessment of statistical downscaling methods for hydrological climate change impact studies. Environ. Model. Softw. 95, 22–28. http://dx.doi.org/10.1016/j.envsoft.2017.03.036.

He, X., Chaney, N.W., Schleiss, M., Sheffield, J., 2016. Spatial downscaling of precipitation using adaptable random forests. Water Resour. Res. 52 (10), 8217–8237.

Hernanz, A., García-Valero, J.A., Domínguez, M., Ramos-Calzado, P., Pastor-Saavedra, M.A., Rodríguez-Camino, E., 2022. Evaluation of statistical downscaling methods for climate change projections over Spain: present conditions with perfect predictors. Int. J. Climatol. 42 (2), 762–776.

Hewitson, B.C., Crane, R.G., 1996. Climate downscaling: Techniques and application. Clim. Res. 7 (2), 85–95.

Hewitson, B.C., Crane, R.G., 2006. Consensus between GCM climate change projections with empirical downscaling: Precipitation downscaling over South Africa. Int. J. Climatol. 26 (10), 1315–1337. http://dx.doi.org/10.1002/joc.1314.

Hotelling, H., 1933. Analysis of a complex of statistical variables into principal components. J. Educ. Psychol. 24 (6), 417.

Hutengs, C., Vohland, M., 2016. Downscaling land surface temperatures at regional scales with random forest regression. Remote Sens. Environ. 178, 127–141. http://dx.doi.org/10.1016/j.rse.2016.03.006.

Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., Iredell, M., Saha, S., White, G., Woollen, J., Zhu, Y., Chelliah, M., Ebisuzaki, W., Higgins, W., Janowiak, J., Mo, K.C., Ropelewski, C., Wang, J., Leetmaa, A., Reynolds, R., Jenne, R., Joseph, D., 1996. The NCEP/NCAR 40-year reanalysis project. Bull. Am. Meteorol. Soc. 77 (3), 437–472. http://dx.doi.org/10.1175/1520-0477(1996)077<0437:TNYRP>2.0.CO;2.

Karl, T.R., Wang, W.-C., Schlesinger, M.E., Knight, R.W., Portman, D., 1990. A method of relating general circulation model simulated climate to the observed local climate. Part I: Seasonal statistics. J. Clim. 3 (10), 1053–1079.

Katz, R.W., Brown, B.G., 1992. Extreme events in a changing climate: Variability is more important than averages. Clim. Change 21 (3), 289–302.

Kilsby, C.G., Jones, P., Burton, A., Ford, A., Fowler, H.J., Harpham, C., James, P., Smith, A., Wilby, R., 2007. A daily weather generator for use in climate change studies. Environ. Model. Softw. 22 (12), 1705–1719. http://dx.doi.org/10.1016/j.envsoft.2007.02.005.

Knutti, R., Sedláček, J., 2013. Robustness and uncertainties in the new CMIP5 climate model projections. Nature Clim. Change 3 (4), 369–373.

Kohonen, T., 1990. The self-organizing map. Proc. IEEE 78 (9), 1464–1480. http://dx.doi.org/10.1109/5.58325.

Koutroulis, A.G., Grillakis, M., Tsanis, I., Papadimitriou, L., 2016. Evaluation of precipitation and temperature simulation performance of the CMIP3 and CMIP5 historical experiments. Clim. Dynam. 47 (5), 1881–1898. http://dx.doi.org/10.1007/s00382-015-2938-x.

Kysel, J., et al., 2002. Comparison of extremes in GCM-simulated, downscaled and observed central-European temperature series. Clim. Res. 20 (3), 211–222.

Li, K.-C., 1991. Sliced inverse regression for dimension reduction. J. Amer. Statist. Assoc. 86 (414), 316–327. http://dx.doi.org/10.1080/01621459.1991.10475035.

Ma, Y., Zhu, L., 2013. A review on dimension reduction. Internat. Statist. Rev. 81 (1), 134–150.

Maraun, D., 2013. Bias correction, quantile mapping, and downscaling: Revisiting the inflation issue. J. Clim. 26 (6), 2137–2143.

Maraun, D., Wetterhall, F., Ireson, A., Chandler, R., Kendon, E., Widmann, M., Brienen, S., Rust, H., Sauter, T., Themeßl, M., et al., 2010. Precipitation downscaling under climate change: Recent developments to bridge the gap between dynamical models and the end user. Rev. Geophys. 48 (3).

Maraun, D., Widmann, M., 2018. Statistical Downscaling and Bias Correction for Climate Research. Cambridge University Press.

Massey, Jr., F.J., 1951. The Kolmogorov-Smirnov test for goodness of fit. J. Amer. Statist. Assoc. 46 (253), 68–78.

Mehran, A., AghaKouchak, A., Phillips, T.J., 2014. Evaluation of CMIP5 continental precipitation simulations relative to satellite-based gauge-adjusted observations. J. Geophys. Res.: Atmos. 119 (4), 1695–1707.

Murphy, J., 1999. An evaluation of statistical and dynamical techniques for downscaling local climate. J. Clim. 12 (8), 2256–2284. http://dx.doi.org/10.1175/1520-0442(1999)012<2256:AEOSAD>2.0.CO;2.

Najafi, M.R., Moradkhani, H., Wherry, S.A., 2011. Statistical downscaling of precipitation using machine learning with optimal predictor selection. J. Hydrol. Eng. 16 (8), 650–664.

National Climatic Data Center, 2020. Global summary of day. URL https://catalog.data.gov/dataset/global-surface-summary-of-the-day-gsod.

Ning, L., Mann, M.E., Crane, R., Wagener, T., 2012. Probabilistic projections of climate change for the mid-Atlantic region of the United States: Validation of precipitation downscaling during the historical era. J. Clim. 25 (2), 509–526.

Pang, B., Yue, J., Zhao, G., Xu, Z., 2017. Statistical downscaling of temperature with the random forest model. Adv. Meteorol. 2017, http://dx.doi.org/10.1155/2017/7265178.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in python. J. Mach. Learn. Res. 12, 2825–2830.

Perkins, S.E., Pitman, A., Holbrook, N., McAneney, J., 2007. Evaluation of the AR4 climate models' simulated daily maximum temperature, minimum temperature, and precipitation over Australia using probability density functions. J. Clim. 20 (17), 4356–4376. http://dx.doi.org/10.1175/JCLI4253.1.

Perkins, S., Pitman, A., Sisson, S., 2013. Systematic differences in future 20 year temperature extremes in AR4 model projections over Australia as a function of model skill. Int. J. Climatol. 33 (5), 1153–1167.

Peterson, T.C., 2005. Climate change indices. WMO Bull. 54 (2), 83–86.

Pierce, D.W., Cayan, D.R., Thrasher, B.L., 2014. Statistical downscaling using localized constructed analogs (LOCA). J. Hydrometeorol. 15 (6), 2558–2585.

Polasky, A.D., Evans, J.L., Fuentes, J.D., Hamilton, H.L., 2021. Statistical climate model downscaling for impact projections in the midwest United States. Int. J. Climatol..

Priestley, M.D.K., Ackerley, D., Catto, J.L., Hodges, K.I., McDonald, R.E., Lee, R.W., 2020. An overview of the extratropical storm tracks in CMIP6 historical simulations. J. Clim. 33 (15), 6315–6343. http://dx.doi.org/10.1175/JCLI-D-19-0928.1.

Quintero, F., Mantilla, R., Anderson, C., Claman, D., Krajewski, W., 2018. Assessment of changes in flood frequency due to the effects of climate change: Implications for engineering design. Hydrology 5 (1), 19.

Radić, V., Clarke, G.K., 2011. Evaluation of IPCC models' performance in simulating late-twentieth-century climatologies and weather patterns over north America. J. Clim. 24 (20), 5257–5274. http://dx.doi.org/10.1175/JCLI-D-11-00011.1.

Robinson, P., Finkelstein, P., 1991. The development of impact-oriented climate scenarios. Bull. Am. Meteorol. Soc. 72 (4), 481–490.

Sa'adi, Z., Shahid, S., Chung, E.-S., bin Ismail, T., 2017. Projection of spatial and temporal changes of rainfall in Sarawak of Borneo island using statistical downscaling of CMIP5 models. Atmos. Res. 197, 446–460. http://dx.doi.org/10.1016/j.atmosres.2017.08.002.

Schlenker, W., Roberts, M.J., 2009. Nonlinear temperature effects indicate severe damages to U.S. crop yields under climate change. Proc. Natl. Acad. Sci. 106 (37), 15594–15598. http://dx.doi.org/10.1073/pnas.0906865106.

Sinha, P., Mann, M.E., Fuentes, J.D., Mejia, A., Ning, L., Sun, W., He, T., Obeysekera, J., 2018. Downscaled rainfall projections in south florida using self-organizing maps. Sci. Total Environ. 635, 1110–1123. http://dx.doi.org/10.1016/j.scitotenv.2018.04.144.

Stephens, G.L., L'Ecuyer, T., Forbes, R., Gettelmen, A., Golaz, J.-C., Bodas-Salcedo, A., Suzuki, K., Gabriel, P., Haynes, J., 2010. Dreary state of precipitation in global models. J. Geophys. Res.: Atmos. 115 (D24), http://dx.doi.org/10.1029/2010JD014532.

Taylor, K.E., Stouffer, R.J., Meehl, G.A., 2012. An overview of CMIP5 and the experiment design. Bull. Am. Meteorol. Soc. 93 (4), 485–498. http://dx.doi.org/10.1175/BAMS-D-11-00094.1.

Teegavarapu, R.S., Goly, A., 2018. Optimal selection of predictor variables in statistical downscaling models of precipitation. Water Resour. Manage. 32 (6), 1969–1992.

Timbal, B., Hope, P., Charles, S., 2008. Evaluating the consistency between statistically downscaled and global dynamical model climate change projections. J. Clim. 21 (22), 6052–6059.

Uriarte, E.A., Martín, F.D., 2005. Topology preservation in SOM. Int. J. Appl. Math. Comput. Sci. 1 (1), 19–22.

Wang, X., Huang, G., Lin, Q., Nie, X., Cheng, G., Fan, Y., Li, Z., Yao, Y., Suo, M., 2013. A stepwise cluster analysis approach for downscaled climate projection–a Canadian case study. Environ. Model. Softw. 49, 141–151.

Wang, J., Nathan, R., Horne, A., Peel, M.C., Wei, Y., Langford, J., 2017. Evaluating four downscaling methods for assessment of climate change impact on ecological indicators. Environ. Model. Softw. 96, 68–82.

Wigley, T.M., 1985. Impact of extreme events. [impact of climate variations]. 316.

Wilby, R.L., Dawson, C.W., 2013. The statistical downscaling model: Insights from one decade of application. Int. J. Climatol. 33 (7), 1707–1719.

Wilks, D.S., 1999. Multisite downscaling of daily precipitation with a stochastic weather generator. Clim. Res. 11 (2), 125–136.

Wood, A.W., Leung, L.R., Sridhar, V., Lettenmaier, D., 2004. Hydrologic implications of dynamical and statistical approaches to downscaling climate model outputs. Clim. Change 62 (1–3), 189–216.

Yang, X., Huang, P., 2022. Improvements in the relationship between tropical precipitation and sea surface temperature from CMIP5 to CMIP6. Clim. Dynam. 1–19.

Zamani, Y., Hashemi Monfared, S.A., Azhdari Moghaddam, M., Hamidianpour, M., 2020. A comparison of CMIP6 and CMIP5 projections for precipitation to observational data: The case of northeastern Iran. Theor. Appl. Climatol. 142, 1613–1623.

Zhao, T., Bennett, J.C., Wang, Q., Schepen, A., Wood, A.W., Robertson, D.E., Ramos, M.-H., 2017. How suitable is quantile mapping for postprocessing GCM precipitation forecasts? J. Clim. 30 (9), 3185–3196.