UN environment

University of Natural Resources
and Life Sciences, Vienna

WITH FUNDING FROM
AUSTRIAN DEVELOPMENT COOPERATION

Ensemble of Bias Corrected Climate Change Scenarios and Indicators for the Western Balkan Region

User Guide

Scenario Download

Climate Indicators

Model-Selection Tool V1.3

ICC-OBS Tool V1.3

Downscaling Tool V1.3

CLIMA PROOF

Version 2.1

Vienna, October 2022

This document presents the ensemble of bias-corrected climate scenarios for the Wester Balkan region as well as the tools and climate indicators developed within the ClimaProof[1] project.

It includes:

- General information on the ensemble of bias corrected climate scenarios developed within the ClimaProof project
- The climate indicators developed within the ClimaProof project
- Instructions to access and download the climate scenarios and indicators
- Step-by-step instructions on the use of the tools developed within the ClimaProof project
    - ClimaProof Model Selection Tool
    - ICC-OBS Tool for integrating own observational data
    - ClimaProof Downscaling Tool for downscaling the climate scenarios to a 0.01° grid
- Instructions to calculate the climate indicators yourself
- A use case that integrates the tools and data developed in the project in a practical example

The scenarios are available for download via the CCCA Data Centre: https://data.ccca.ac.at/group/climaproof

The tools, documents and scripts are provided via the BOKU-Met GitHub repository: https://github.com/boku-met

A document with details on the methodical background of the scenarios and tools is provided in addition to this user guide (available for download via https://github.com/boku-met/climaproof-docs)

Please note that the materials described below are in their Beta phase and therefore bugs can occur. The description presented reflects the status quo (Version 1.3, status as of May 2022). In the course of further development, modifications and changes might occur. The document will be updated accordingly.

Citation:

Formayer, H.; Wind, M.; König, B.; Becsi, B., 2022. High Resolution Climate Change Projections and Indicators. User Guide. Project ClimaProof. Version 2.1: October 2022.

Heading project coordination:

Harald Egerer, Ana Vukoje and Sonja Gebert
Vienna Programme Office, UN Environment

[1] For more information on the project please see page 5 or https://github.com/boku-met/climaproof-docs

Table of Contents

The ClimaProof project

The project **“Enhancing Environmental Performance and Climate Proofing of Infrastructure Investments in the Western Balkan Region from an EU integration perspective” (ClimaProof)** is financed by the Austrian Development Agency and implemented by the United Nations Environmental Programme. ClimaProof will result in increased technical capacities of the relevant national authorities in the field of climate change adaptation, specifically climate proofing of road infrastructure, green infrastructure and evidence-based policy development in the field of climate change adaptation.

Furthermore, it aims at raising awareness of the relevant government officials in the Western Balkan region in regard to climate change in general, and in particular on climate change impacts on road infrastructure and the specific needs of the infrastructure sectors for increased resilience.

This will be achieved through the development of a regional strategy on climate resilient infrastructure, tailor-made training modules and development of guidelines, enhanced dialogue via regional communication tools as well as exchange of information, experiences and best practices via networking and training events.

The first component of the ClimaProof project focuses on **understanding the future climate and weather patterns in the target region**. This includes strengthening national capacities to understand climate change and climate change related risks in the region through **improvement of the information base.**

In order to do so the Institute of Meteorology and Climatology of the University of Natural Resources and Life Sciences, Vienna (BOKU-Met) developed

1. an **ensemble of bias corrected climate change scenarios and climate indicators for the Western Balkan Region**
2. an **easy to handle tool for bias correction** for “Improving bias-corrected Climate Change scenarios with local OBServational data” (**ICC-OBS Tool**),
3. a **model selection tool**
4. a **downscaling tool**

The climate change scenarios, climate indicators as well as the tools and scripts are freely available.

A step-by-step instruction on the handling of the developed materials is presented in this User Guide. Additionally, it provides a use case that ties all the materials developed in the project together to give a comprehensive overview.

Ensemble of bias-corrected climate scenarios

Features

temporal resolution:	daily
spatial resolution:	0.1°or (0.01° on request)
temporal extent:	1981 – 2099/2100 (depending on the model)
data format:	netCDF
geographic extent:	Western Balkan Region (Albania, Bosnia and Herzegovina, Croatia, Kosovo*, Montenegro, North Macedonia, Serbia) (see Figure 1)

* Reference to Kosovo shall be understood to be in the context of Security Council Resolution UNSCR 1244/99

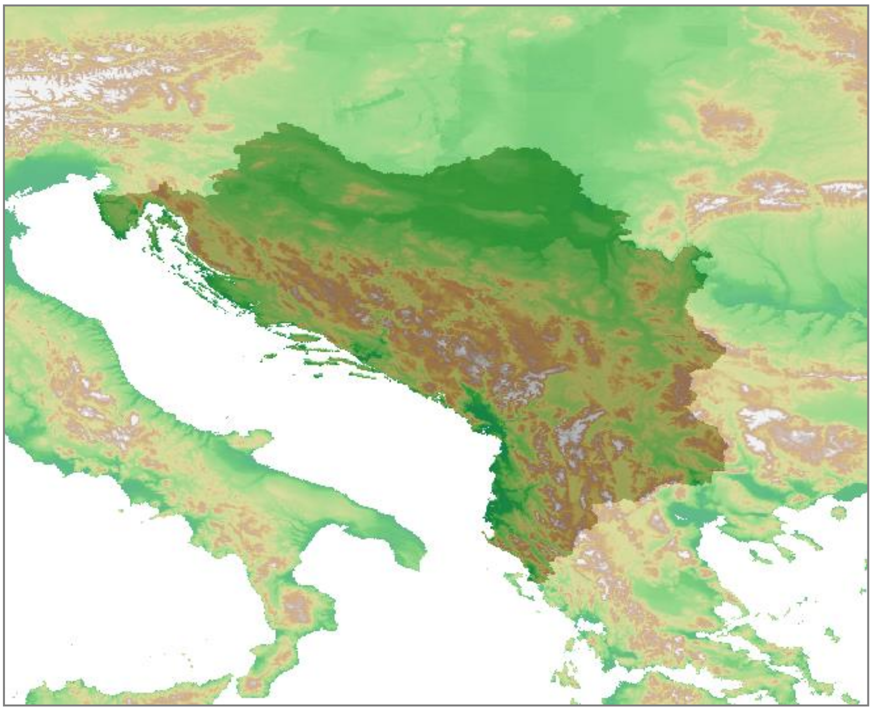

Figure 1: Map of the area covered by the ensemble of bias-corrected climate scenarios (illustrated by BOKU-Met, Datasource: ESRI, Diva-gis)

Climate parameters

Table 1 shows the parameters included in the ClimaProof data set. Apart from the standard parameters temperature and precipitation, additional parameters were chosen to be included in the dataset: radiation, 10-m wind speed and relative humidity. This set of variables is adequate for calculating derived variables like evapotranspiration and providing necessary input to climate change impact models.

Table 1: Climatological parameters included in the bias-corrected scenarios

Parameter	**Unit**	**Description**
tasmax	°C	daily maximum near-surface air temperature
tasmin	°C	daily minimum near-surface air temperature
pr	mm/day	total daily precipitation
rsds	W/m²	daily mean surface downwelling shortwave radiation
sfcWind	m/s	daily mean near-surface wind speed
hurs	%	daily mean near-surface relative humidity

Available models

The ensemble of bias-corrected climate scenarios for the Western Balkan Region produced within the ClimaProof project is based on freely available climate model data (RCM scenarios) from EURO-CORDEX[2] and MED-CORDEX (see Ruti *et al.*, 2016) experiments. EURO-CORDEX and MED-CORDEX scenarios can be downloaded via the online platforms http://www.euro-cordex.net/ or https://www.medcordex.eu/.

The models that are bias-corrected within the ClimaProof project were selected based on the following criteria

- availability (only freely available scenarios are used (status as of April 2018))
- domain (models covering the Western Balkan Region)
- horizontal grid-resolution of 0.11°

The models are run with the RCP (Representative Concentration Pathway) scenario set[3].

All EURO-CORDEX models used are available for the RCP 4.5 and 8.5 scenarios– five of them are additionally available for the RCP 2.6 scenario.
Of the MED-CORDEX models with a resolution of 0.11°, one model uses the RCP 4.5 scenario and two models the RCP 8.5 scenario.
A full list of the models is included in the Annex.

The bias-corrected climate scenarios developed within the ClimaProof project are **freely available** via the **CCCA Data Centre**: https://data.ccca.ac.at/group/climaproof

Every model was bias corrected separately and is available for download as a netCDF file.
The scenarios are provided on regular grid with a horizontal resolution of 0.1°x0.1°.

If required, the downloaded scenarios can further be downscaled to a 0.01° grid (~1 km) using the ClimaProof Downscaling Tool (for more details see chapter The ClimaProof Downscaling Tool).

[2] We acknowledge the World Climate Research Programme's Working Group on Regional Climate, and the Working Group on Coupled Modelling, former coordinating body of CORDEX and responsible panel for CMIP5. We also thank the climate modelling groups (listed in Table 2 of this paper) for producing and making available their model output. We also acknowledge the Earth System Grid Federation infrastructure an international effort led by the U.S. Department of Energy's Program for Climate Model Diagnosis and Intercomparison, the European Network for Earth System Modelling and other partners in the Global Organisation for Earth System Science Portals (GO-ESSP).
3 van Vuuren, D.P., Edmonds, J., Kainuma, M., Riahi, K., et al. (2011) The representative concentration pathways: an overview. Climatic Change. [Online] 109 (1–2), 5–31. Available from: doi:10.1007/s10584-011-0148-z

Additionally, own (higher resolved) observational data can be used to further improve the dataset by using the ICC-OBS Tool developed by BOKU-Met (detailed description see chapter The ICC-OBS Tool).

Bias correction

The climate projections are bias-corrected with open and freely available observational data. The selection of the data sets used is based on the criteria ‘quality’ and ‘resolution’.

Table 2Table 2: Datasets of observational data used for bias-correction, as well as Figure 2 and Figure 3 give an overview on the observational data used for bias correction.

Bias correction was performed with the **Scaled Distribution Mapping (SDM)** method, described in the publication of Switanek *et al*. (2017).

A detailed information on the production of the ensemble of bias-corrected climate scenarios can be found in:

Formayer, H.; Wind, M.; König, B.; Becsi, B., 2022. ClimaProof - Ensemble of Bias-Corrected Climate Change Scenarios and Indicators. Methodical Background V3.0. Project ClimaProof - Deliverable 1.2.2

The document is available for download via https://github.com/boku-met/climaproof-docs

Table 2: Datasets of observational data used for bias-correction

Dataset	Variables used within the ClimaProof project	Resolution	Expansion	Download
E-OBS (Haylock *et al*., 2008; ECA&D, 2018)	tmax, tmin	0.25° x 0.25°	Lat. 25°N -75°N Long. 40°W-75°E	https://www.ecad.eu/download/ensembles/download.php
CARPATCLIM (Szalai et al., 2013; European Commission - JRC, 2013)	tmax, tmin, prec, rg (global radiation)	0.1° x 0.1 °	Lat. 44°N - 50°N, Long. 17°E - 27°E	http://www.carpatclim-eu.org/pages/download/default.aspx
DANUBECLIM (Szalai *et al*., 2013; European Commission - JRC, 2015)	tmax, tmin, prec, rg (global radiation)	0.1° x 0.1 °	Serbia, Montenegro and the Srpska Republic	http://www.carpatclim-eu.org/danubeclim
ERA-5 (ECMWF, 2016:p.7)	sfcWind, rh	0.28° (31km)	global	http://apps.ecmwf.int/data-catalogues/era5/?class=ea
CHIRPS (Funk *et al*., 2015)	pr	0.05° x 0.05°	Lat. 50°N - 50°S, Lon. 180°W - 180°E	http://dx.doi.org/10.15780/G2RP4Q
SARAH-2 (Pfeifroth *et al*., 2017)	SIS (Surface incoming shortwave radiation)	0.05° x 0.05°	Lat. 65°N - 65°S, Long. 65°W - 65°E	https://doi.org/10.5676/EUM_SAF_CM/SARAH/V002

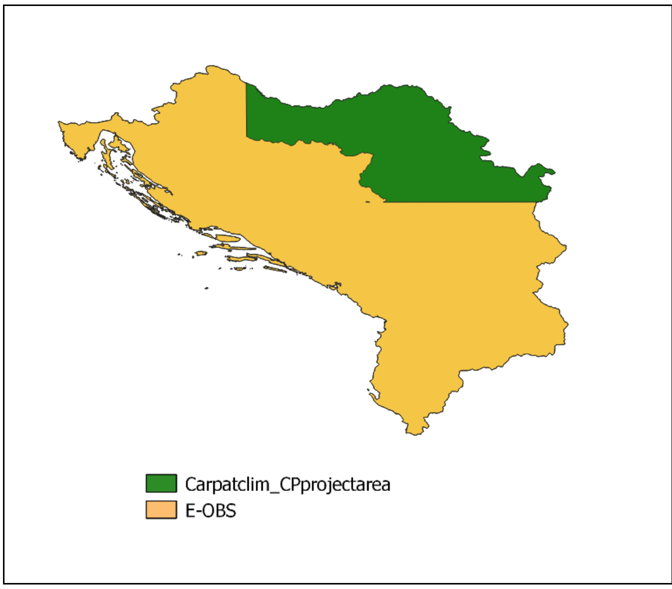

Figure 2: Observational Data used for Bias Correction of Temperature;
(Illustrated by BOKU-Met, Datasource: Diva-gis, Carpatclim, E-OBS)

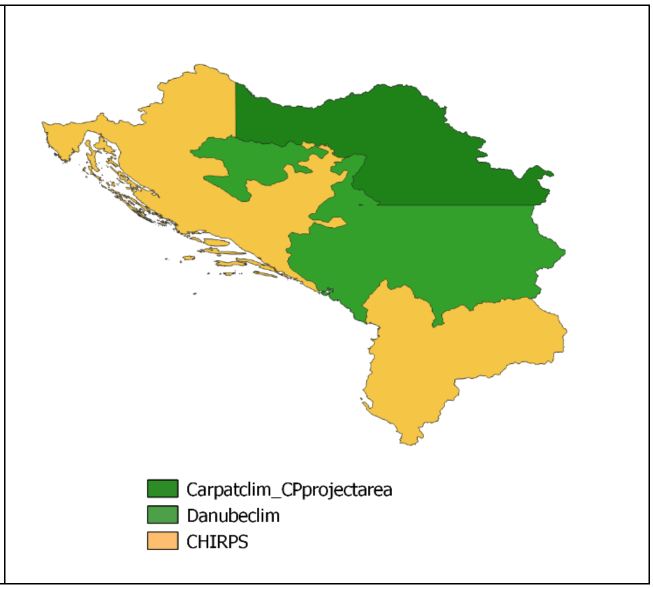

Figure 3: Observational Data used for Bias Correction of Precipitation;
(Illustrated by BOKU-Met, Datasource: Diva-gis, Carpatclim, Danubeclim, CHIRPS)

Climate indicators

In addition to the meteorological parameters, several climate indicators were computed. The indicators were determined in an online survey with experts and authorities working in the fields of infrastructure development, planning, construction, maintaining and operating. The climate indicators translate the raw climate parameters into impacts and risks that are relevant for infrastructure. They are derived directly from the climate model parameters without including any additional data and therefore only present an approximation for the respective impacts. An example is the indicator “Landslides”: It represents a precursor condition for landslides in the form of precipitation intensity thresholds but does not include data on the slope of the terrain, the geological composition of the ground, or soil moisture. Table 3 lists the climate indicators developed in the ClimaProof project and their definitions.

Table 3: Climate indicators developed in the project

Indicator name	Climate Parameter	Definition	Timesteps
Droughts	**Pr**	*Consecutive 5 and 7-day periods of less than 1mm daily precipitation during April-September (days in periods, maximum period length)*	**Annual**
Droughts & Heat	**Pr, Tmax**	*Days in 5 and 7-day drought periods with exceedance of 30°C daily maximum temperature*	**Annual**
Extreme Wind	**Wind**	*Days with exceedance of the 99.9-percentile of daily mean wind speeds of the observation period 1981-2010*	**Annual**
Freeze-Thaw-Cycles	**Tmin, Tmax**	*Days with Tmin ≤ -2.2 °C and Tmax ≥ 0°C*	**Monthly**
Heat Days	**Tmax**	*Days with exceedance of 30°C / 40°C daily maximum temperature*	**Monthly**
Landslides	**Pr**	*Days with exceedance of precipitation intensity thresholds after the Moser-Hohensinn approach (M = 41.66 * (h-0.77) * h) for h = 24, 48 and 72 hours*	**Annual**
Precipitation intensity	**Pr**	*99.9 percentile of daily precipitation sums (3-yearly event)*	**Annual for 30 year periods**
Snowfall	**Pr, Tmin, Tmax**	*Days with precipitation sums of more than 1mm / 10mm and daily mean temperature <= 0°C*	**Monthly**

Download data from the CCCA Data Centre

The CCCA Data Centre – a unit of the CCCA (Climate Change Center Austria) - is the Austrian data infrastructure for climate data and information, enabling parties involved to publish and retrieve resources with respect to existing data policies. Its goal is to support interoperability and promote collaboration between different climate science and research communities, reducing data redundancy and loss of data[4].

The CCCA Data Centre can be accessed online via: https://data.ccca.ac.at/

The ClimaProof dataset can be downloaded via the CCCA Data Centre: https://data.ccca.ac.at/group/climaproof

The following datasets are provided:

- Bias-corrected climate change scenarios for all parameters (Table 1, for a full list of models see Table 5 and Table 6 in the Annex)
- Original climate change scenarios for all parameters, regridded to the common grid with 0.1° resolution (to be used within the ICC-OBS Tool)
- Gridded observational data for all parameters (used for bias correction)
- Climate indicators for all climate change scenarios
- Ensemble means of climate indicators for the three emission scenarios RCP2.6, RCP4.5 and RCP8.5
- Climate indicators for observational data
- topography data of the common grid
- topography data for the high resolution domain (to be used with the ClimaProof Downscaling Tool)
- A link to the ClimaProof tool and scripts collection on GitHub

In order to access the data you have to create an account first. To register go to: https://data.ccca.ac.at/user/register

You will get a confirmation email with a link to finish your registration.

Once you are logged in to the CCCA Data Centre, you can search and filter the data sets (as shown in the screenshot to the right). Among others you can filter by:

- Domain: zoom in our out of the map or draw a rectangle on the map to filter data sets that are available in your location of interest.
- Year: e.g. to find only historical data, select 1981 to 2010.

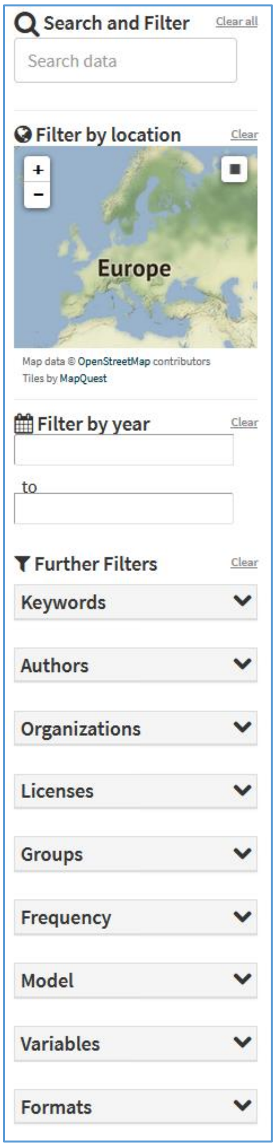

Figure 4: Screenshot of the filtering function of the CCCA Data Centre.

[4] https://ccca.ac.at/ueber-ccca/team/datenzentrum

- Groups: select the ClimaProof group to only show data from the ClimaProof project. With the groups filter you can also select only bias-corrected data, observational data, climate indicators or certain emission scenarios (RCP).
- Model: select a specific climate model of your choice.
- Variables: select the variable you are interested in.
- Keywords: If the data cannot be filtered otherwise, you can provide keywords to find datasets with a certain name (e.g. “regridded” for all datasets with original, not bias-corrected climate model data).

When selecting a data set on the CCCA Data Centre you can explore the metadata information of the data by browsing through the metadata tabs. The data itself is stored as resources that are listed above the metadata (see Figure 5).

By clicking on the “Explore” button, you can get a preview of the data or download the dataset. In the preview you can have a quick look at single time steps of the dataset. Under Map Parameter you can further modify the visualisation by selecting the desired variable or the minimum and maximum value that is shown (see Figure 6).

Attention! The visualization previews are only available after login. Data has been updated due to bugfixes and re-uploaded to the CCCA data centre. Therefore, not all datasets are stored in the same internal server storage but may be stored on external storage connected to the server. **For these datasets, previews and subsetting (clipping the data to spatial and temporal extents) are not available even after login.**

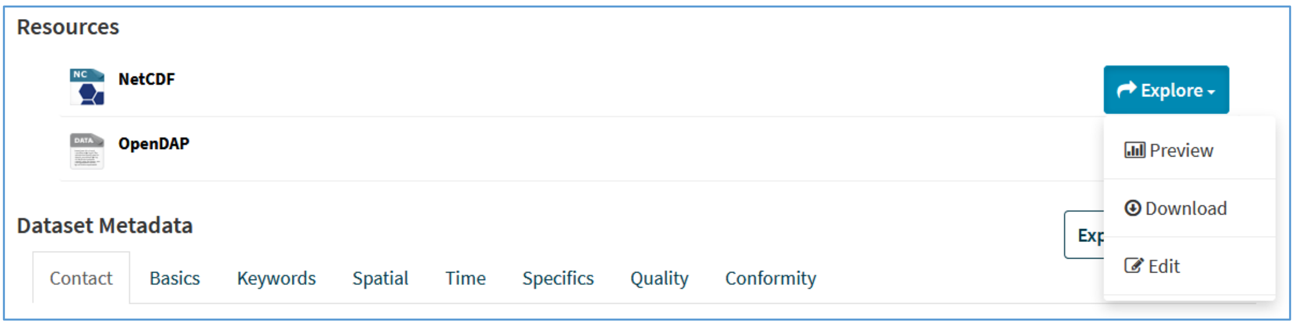

Figure 5: Screenshot ‘Explore’ data on the CCCA Data Centre

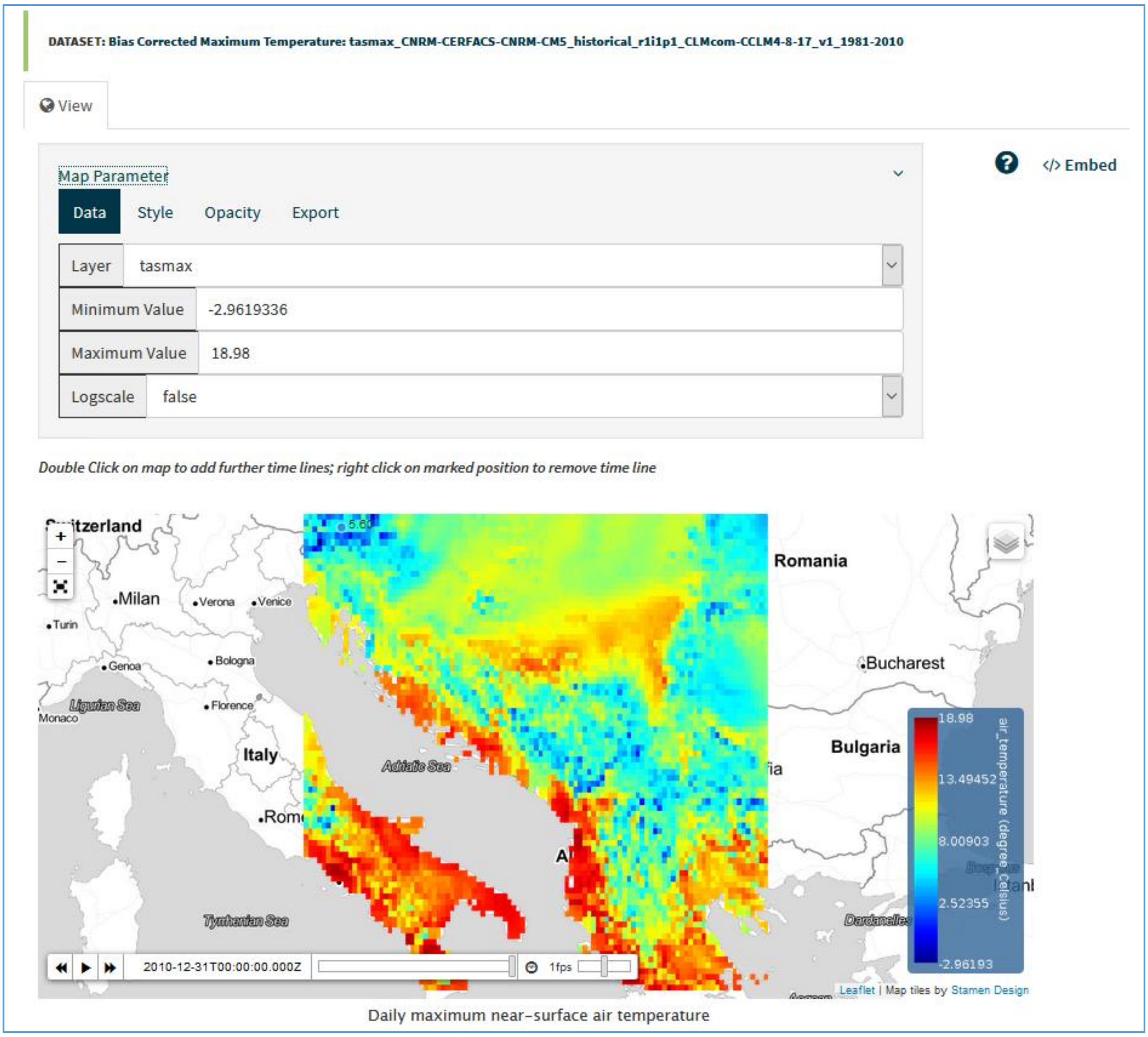

Figure 6: Screenshot Preview data on data.ccca.ac.at.

Installing the Python environment

The materials developed within the ClimaProof project are programmed in Python.

In order to run them on your local machine, you have to set up a Python Environment.

The easiest way to do this is via Miniconda - an open-source package management system that runs on Windows (64-bit), macOS and Linux. Furthermore, Miniconda allows you to install everything in your home-directory without the need for administrative or root permissions.

To set up the Python Environment please follow the step-by-step instruction below.

Notes on the installation:
Please be aware that the Python installation requires approximately 10GB of space on your disk so choose a directory that has enough space available.
Unfortunately, the Downscaling Tool currently only works under Linux and MacOS. As a workaround, Windows 10 users can run the tool through a Docker container. For this, Docker has to be installed on your Computer.

Attention (Miniconda users only)!
In the current versions of the ClimaProof tools (v1_3+), bugfixes in some of the required backend libraries were implemented. In order to update the backend libraries, writing permissions to the installed Miniconda folders by the user are required. The updates are done automatically the first time a tool is started.

Miniconda

Linux

1. Download Miniconda (Python 3) for Linux (https://conda.io/en/latest/miniconda.html):

   You can do that manually by clicking on the link above and downloading the installer with your browser or by opening a terminal and running:

   ```
   wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
   ```

2. Open a Terminal window, navigate to the directory containing the installation file and run:

   ```
   sh Miniconda3-latest-Linux-x86_64.sh
   ```

3. Follow the instructions on the installer screens - accept defaults settings (you can change them later).
4. Close and re-open your Terminal window to make the changes take effect.
   To test your installation, type:

   ```
   conda list
   ```

   If your installation was successful, a list of installed packages appears.
5. Update Conda:

```
conda update conda
```

Windows

1. Download Miniconda (Python 3) for Windows (https://conda.io/en/latest/miniconda.html)
2. Start the .exe file and follow the instructions of the installer or
   open the Anaconda Prompt, navigate to the directory containing the installation file

   ```
   cd directory
   ```

   e.g.

   ```
   cd C:\Users\Me\Downloads
   ```

   and run:

   ```
   start /wait "" Miniconda3-latest-Windows-x86_64.exe /InstallationType=JustMe /RegisterPython=0 /S
   /D=%UserProfile%\Miniconda3
   ```

3. Open “Anaconda Promt” – a terminal window appears.
   To test your installation, type:

   ```
   conda list
   ```

   If your installation was successful, a list of installed packages appears.
4. Update Conda

   ```
   conda update conda
   ```

MacOS

1. The easiest way to install Miniconda on Mac is via homebrew (see homebrew website for current changes on how to install: https://brew.sh/).
   You can install homebrew by opening a terminal and entering:

   ```
   /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
   ```

2. Once homebrew is installed, you can install Miniconda with

   ```
   brew cask install miniconda
   ```

3. Close and re-open your Terminal window to make the changes take effect.
   To test your installation, type:

   ```
   conda list
   ```

   If your installation was successful, a list of installed packages appears.
4. Update Conda:

   ```
   conda update conda
   ```

Docker

The installation of Docker is only necessary if you want to use the ClimaProof Downscaling Tool under Windows. For the installation you need to have administrative permissions on your computer.

Docker Desktop for Windows runs on 64-bit Windows 10 Pro, Enterprise, and Education (1607 Anniversary update, Build 14393 or later). We currently cannot support older Windows versions.

Docker Desktop for Windows

Please follow the instructions on https://docs.docker.com/docker-for-windows/install to install Docker Desktop for Windows.

- Download Docker
- Install Docker
- Add user to Docker user group
- allow virtualization in BIOS settings

The Model Selection Tool

To assist with the selection of a specific climate change scenario the ClimaProof Model Selection Tool has been developed. It compares and visualizes climate change signals of the raw climate model data.

The climate change signal is calculated by subtracting the climatological mean of the historical period (1981-2010) from the future period (near, mid or far future). The climate change signal for temperature is an absolute value (in °C). For precipitation and radiation, a relative climate change signal (in %) is calculated by dividing the absolute value with the historical mean.

Selection of model(s) - Help for decision making

Ensemble of models for a representative concentration pathway (RCP)

Download of all models available for the chosen RCP (see Table 2 and 3). The output contains all the data for the respective RCP.

This is the best option if you would like to compare the models and know the full range of predicted changes.

Single model

If you do not have the capacities to download and work with the full ensemble, downloading a single model is also an option.

If you already know which model you would like to download you can choose it right away or you can use the specially developed "Model Selection Tool", that compares the models according to your parameters and can thus help you to choose the best model for your needs (step-by-step guide see below).

Installation

Linux/MacOS

1. Download the ClimaProof Toolbox from the BOKU-Met GitHub repository. Go to https://github.com/boku-met/climaproof-tools, download the tools via your browser and unzip it. The Toolbox also includes the Downscaling Tool.
2. Open a terminal and navigate to the *climaproof-tools-master* directory

```
cd directory
```

e.g.

```
cd ~/user/climaproof-tools-master
```

3. Create a conda environment (called *tools*)

```
conda env create --name desired_name_of_environment --file environment_tools.yml
```

Attention! Do NOT update this conda environment. The tools have been written and tested for the specific package versions of this environment. Updating it to the latest version will stop the tools from working correctly.

Installation Notes for MacOS:

The package *shapely* that is installed with the conda environment can cause a known issue on MacOS systems when trying to run the ClimaProof Tools. The module tries to load the geos library and fails, causing an error of the form 'OSError: Could not find lib c or load any of its variants []'.

If you encounter such an error, there is a workaround (reported here):

You need to set the environment variable DYLD_FALLBACK_LIBRARY_PATH to the standard library directories on your system. To do that, just add the line

```
export DYLD_FALLBACK_LIBRARY_PATH=$HOME/lib:/usr/local/lib:/lib:/usr/lib:$DYLD_FALLBACK_LIBRARY_PATH
```

to your shell's configuration file (usually a "."-file in your home directory, e.g. *.bashrc*, *.zshrc*).

ATTENTION: csh and other shell variants may use a different syntax to change environmental variables. You can find your shell version by typing

```
echo $SHELL
```

into your Terminal. Finding the syntax and the source file for changing environmental variables should be easy once you know your shell version.

Don't forget to close and reopen your terminal to make the change take effect.

To check if the environment variable has been set correctly, after reopening the terminal type:

```
echo $DYLD_FALLBACK_LIBRARY_PATH
```

The output should look like this:

```
/Users/{user}/lib:/usr/local/lib:/lib:/usr/lib:
```

Windows

1. Download the ClimaProof Toolbox from the BOKU-Met GitHub repository. Go to https://github.com/boku-met/climaproof-tools, download the tools via your browser and unzip it. The Toolbox also includes the Downscaling Tool.
2. Open a "Anaconda Prompt" window and navigate to the climaproof-tools directory

```
cd directory
```

e.g.

```
cd C:\Users\Me\climaproof-tools-master
```

3. Create a conda environment (called *tools*)

```
conda env create --name desired_name_of_environment --file environment_tools.yml
```

Attention! Do NOT update this conda environment. The tools have been written and tested for the specific package versions of this environment. Updating it to the latest version will stop the tools from working correctly.

Running the Tool

Linux/MacOS

1. In your Terminal activate the conda environment:

```
conda activate desired_name_of_environment
```

2. Navigate to the *climaproof-tools* directory and type in your command line:

```
cd climaproof-tools-master

bokeh serve --show mst
```

This will start the Model Selection Tool in a new tab of your browser.

Windows

1. In your “Anaconda Prompt” window activate the conda environment:

```
activate desired_name_of_environment
```

2. Navigate to the *climaproof-tools-master* directory and type in your command line:

```
bokeh serve --show mst
```

This will start the Model Selection Tool in a new tab in your browser.

Quitting the Tool

Once you are finished with model selection, you can close the browser Tab. However, this does not terminate the Bokeh server running in the background. In your Terminal, Press CTRL + c to shut down the running server.

Model Selection tool - Step-by-Step guide

In your browser the Tool will look like shown in Figure 7.

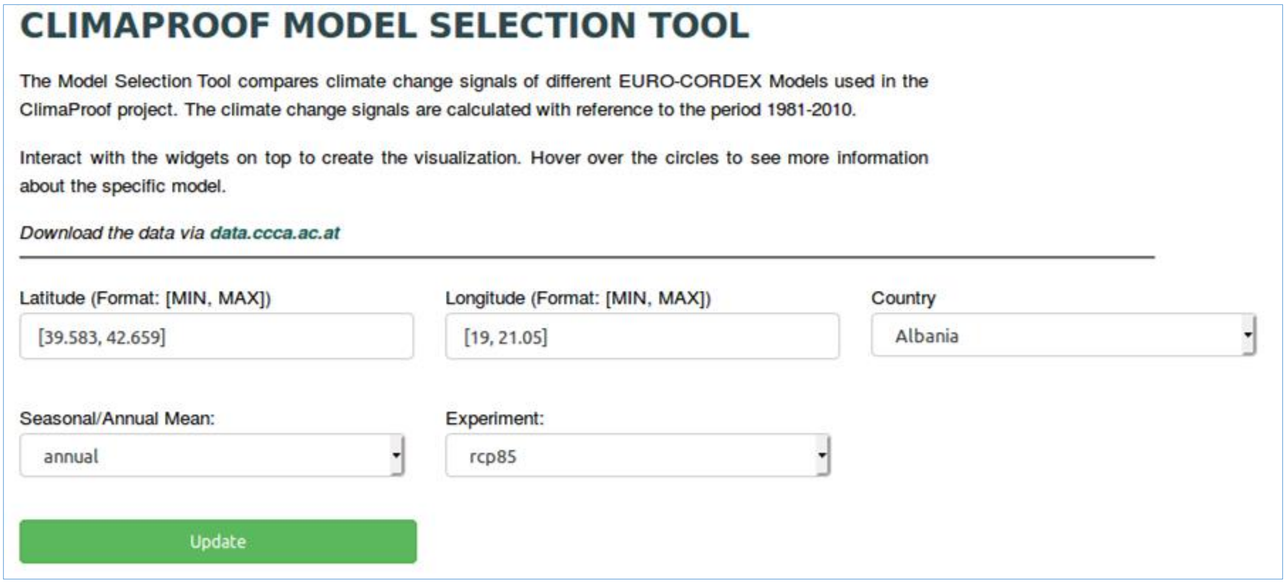

Figure 7: Screenshot of the ClimaProof Model Selection Tool - Header

Choose the parameters required:

- Select the bounding box of the desired domain. Type in latitude and longitude (in decimal degrees) or select a country. Latitude and longitude are automatically updated if you choose a country from the drop-down menu.
- Choose if you want an annual, summer (JJA) or winter (DJF) mean of the climate change signal.
- Select the RCP scenario that you are interested in (2.6, 4.5 or 8.5) - a more detailed explanation of the RCPs is provided in the Methodical Background document.
- Click on the "Update" button to create the visualization. This takes a few seconds to compute.

Once the visualization (see Figure 8) is created you can modify it according to your specific interest:

- Choose the parameters that should be shown on the X- and Y-Axis. Available parameters are:
  - precipitation (pr)
  - maximum and minimum temperature (tasmax, tasmin)
  - global radiation (rsds)
- Choose the future time frame of interest:
  - Near future (2021-2050)
  - Mid of century (2036-2065)
  - Far future (2070-2099)

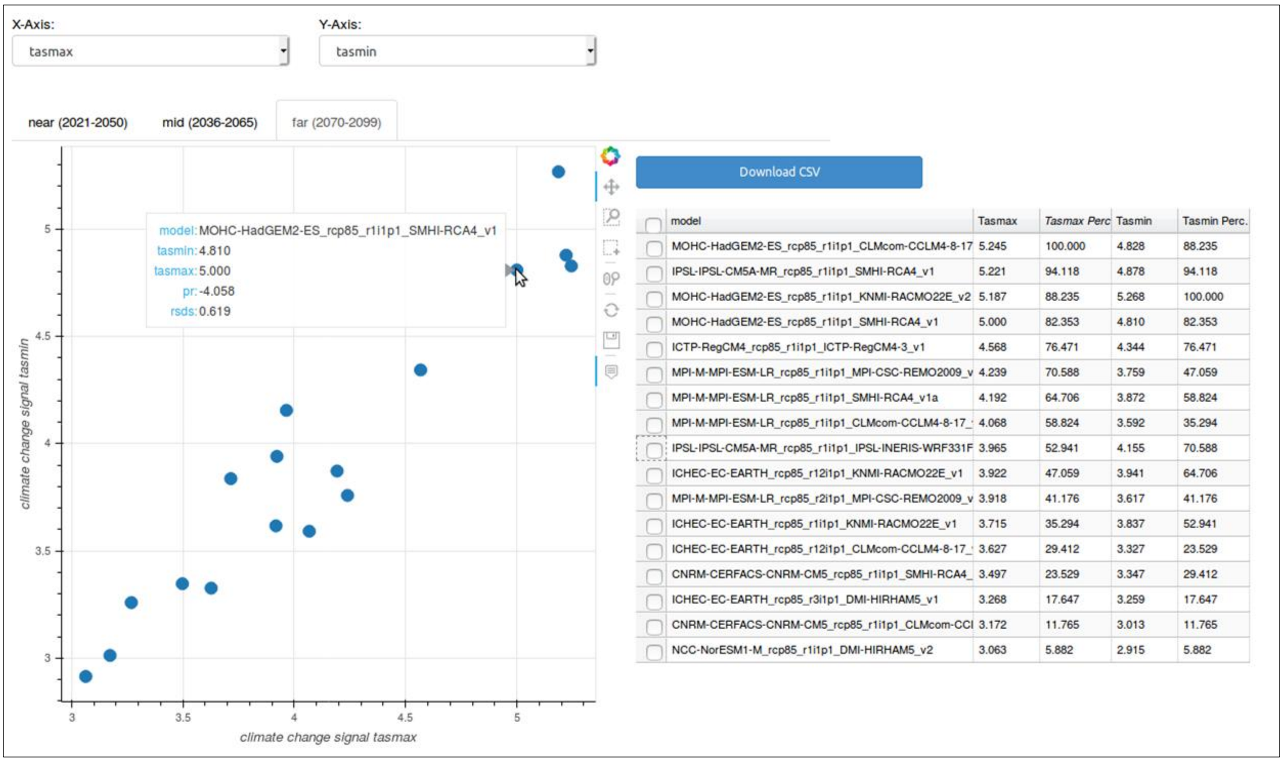

Figure 8: Screenshot of the ClimaProof Model Selection Tool - Visualization.

Find out more about the model and the related climate change signals by hovering over the points in the visualization. You can zoom in, select points with a custom drawn box or export the plot as a png.

As additional information, a table is created. It shows the climate change signal and the corresponding percentiles for the selected parameters.

> If you want to find a model representing the median of all the models (for the specific region, time frame and variable), look up the model in the table that is nearest to the 50% percentile.

The ICC-OBS Tool

*"**I**mproving bias-corrected **C**limate **C**hange scenarios with local **OBS**ervational data"*

The ICC-OBS tool allows the integration of additional local observational data for further improvement of the climate change scenarios provided.

The main steps, the ICC-OBS Tool performs are:

- Selection of the area of interest (by latitude and longitude)
- Integration of additional observational data (station data) to the existing gridded observations to improve local observation quality
- Bias correction with the improved observational dataset and saving the new bias corrected data to a CF-conform netCDF file.

The ICC-OBS tool is programmed in Python and uses netCDF as it's default file format.
A detailed description of the methods used within the tool is provided in the Methodical Background document (download via: https://github.com/boku-met/climaproof-docs).

Installation

Linux

1. Download the ICC-OBS tool from the ClimaProof GitHub repository (https://github.com/boku-met/ICC-OBS) and unzip it.
2. Navigate to the ICC-OBS directory in your Terminal and create a conda environment - this automatically installs all the python packages you need to run the ICC-OBS tool:

```
cd ICC-OBS-master

conda env create --name desired_name_of_environment --file requirements.yml
```

Attention! Do NOT update this conda environment. The tools have been written and tested for the specific package versions of this environment. Updating it to the latest version will stop the tools from working correctly.

3. Activate the ICC-OBS environment:

```
conda activate desired_name_of_environment
```

Windows

1. Download the ICC-OBS tool from the ClimaProof GitHub repository (https://github.com/boku-met/ICC-OBS)
2. Unpack the zip-file.

3. Open the Anaconda Prompt and navigate to the *ICC-OBS-master* directory (the unzipped folder):

```
cd directory\to\ICC-OBS-master
```

e.g.

```
cd C:\Users\Me\ICC-OBS-master
```

4. Create a conda environment - this automatically installs all the python packages you need to run the ICC-OBS tool:

```
conda env create --name desired_name_of_environment --file requirements.yml
```

5. Activate the ICC-OBS environment:

```
activate desired_name_of_environment
```

6. Install additional packages (sometimes these packages fail to install in the previous steps):

```
pip install gooey==1.0.2

pip install pykrige==1.4.1
```

Attention! Do NOT update this conda environment. The tools have been written and tested for the specific package versions of this environment. Updating it to the latest version might stop the tools from working correctly.

MacOS

Unfortunately, the ICC-OBS Tool does not work reliably under MacOS at this time. You can try to install it similar to the Linux guide, but you might encounter errors that prohibit the tool to be displayed on the screen.

Input data

In order to run the ICC-OBS Tool the following input data is needed:

1. Original climate model data (**not bias corrected**) - available via https://data.ccca.ac.at/group/climaproof
2. Gridded observations - available via https://data.ccca.ac.at/group/climaproof
3. Topography file - available via https://data.ccca.ac.at/group/climaproof (search for “Topography”)
4. Station data (time series and metadata) - your own data

The climate model data serving as input for the ICC-OBS Tool is the **original climate model data**, regridded to the common grid with 0.1° resolution. Due to statistical issues in the bias correction algorithm, using the bias corrected model as input data can lead to wrong results. It is important, that all gridded data (climate model data, gridded observations and topography file) are on the same grid with **the same resolution**.

Station data

In order to seamlessly use additional station data in the ICC-OBS Tool, the data has to be formatted in a specific way. First of all, metadata and station time series have to be saved in separate files. General requirements for the files are:

- Files should be in .csv (text) format (easy to export from excel) or ASCII .txt
- Decimal numbers should be separated with a dot "."
- Columns should be separated with a comma ","
- There should be one file that contains the metadata (station number and name, latitude, longitude and height) for all stations
- The time series data for each station and each variable should be stored in separate files.
- All files should contain a header (first row).

> In order to get improved results compared to the bias corrected scenarios provided within the ClimaProof dataset, **good quality of the station data is important**. The data should be quality controlled and cover ideally all 30 years, but at least 10 years, of the time period 1981-2010.

Metadata files

The metadata file should contain the station number (integer), name, latitude and longitude (in decimal degrees) and height (in meters) for each station.

Example (dummy data for three stations)

```
stationnr,name,lat,lon,height
123,Station_a,45.25,20.3,500
124,Station_b,45.27,21.66,120
125,Station_c,45.23,20.573,210
```

Time Series Files

The time series files should be named 'stationnumber_variable.csv' (e.g. 123_tasmax.csv') and should contain one column for date and one column for the data. Further requirements are:

- Date Format: YYYY-MM-DD
- The temporal frequency of the data should be daily.
- Missing data should be marked with "nan"

Example (dummy data for maximum temperature)

```
time,tasmax
1981-01-01,2.1
1981-01-02,2.9
1981-01-03,1.1
1981-01-04,1.2
1981-01-05,nan
1981-01-06,3.8
```

```
1981-01-07,5.2
```

Running the tool

If you successfully installed Miniconda and created the ICC-OBS environment (see above), you can now start the tool. In order to run it you should have all the data you need already downloaded and ready to use on your local machine.

Attention (Miniconda users only)!

In the current versions of the ClimaProof tools (v1_3+), bugfixes in some of the required backend libraries were implemented. In order to update the backend libraries, writing permissions to the installed Miniconda folders by the user are required. The updates are done automatically the first time a tool is started.

1. Navigate to the ICC-OBS folder containing the file main.py and activate the conda environment:

   Linux:

   ```
   conda activate desired_name_of_environment
   ```

   Windows:

   ```
   activate desired_name_of_environment
   ```

2. Start the tool by running main.py:

   ```
   python main.py
   ```

   This will start the graphical user interface of the ICC-OBS Tool (see Figure 9).

MacOS

Unfortunately, the ICC-OBS Tool does not work reliably under MacOS at this time. You can try to install it similar to the Linux guide, but you might encounter errors that prohibit the GUI of the tool to be displayed on the screen.

ICC-OBS tool - Step-by-Step guide

Once you start the ICC-OBS Tool you can start filling out the required fields.

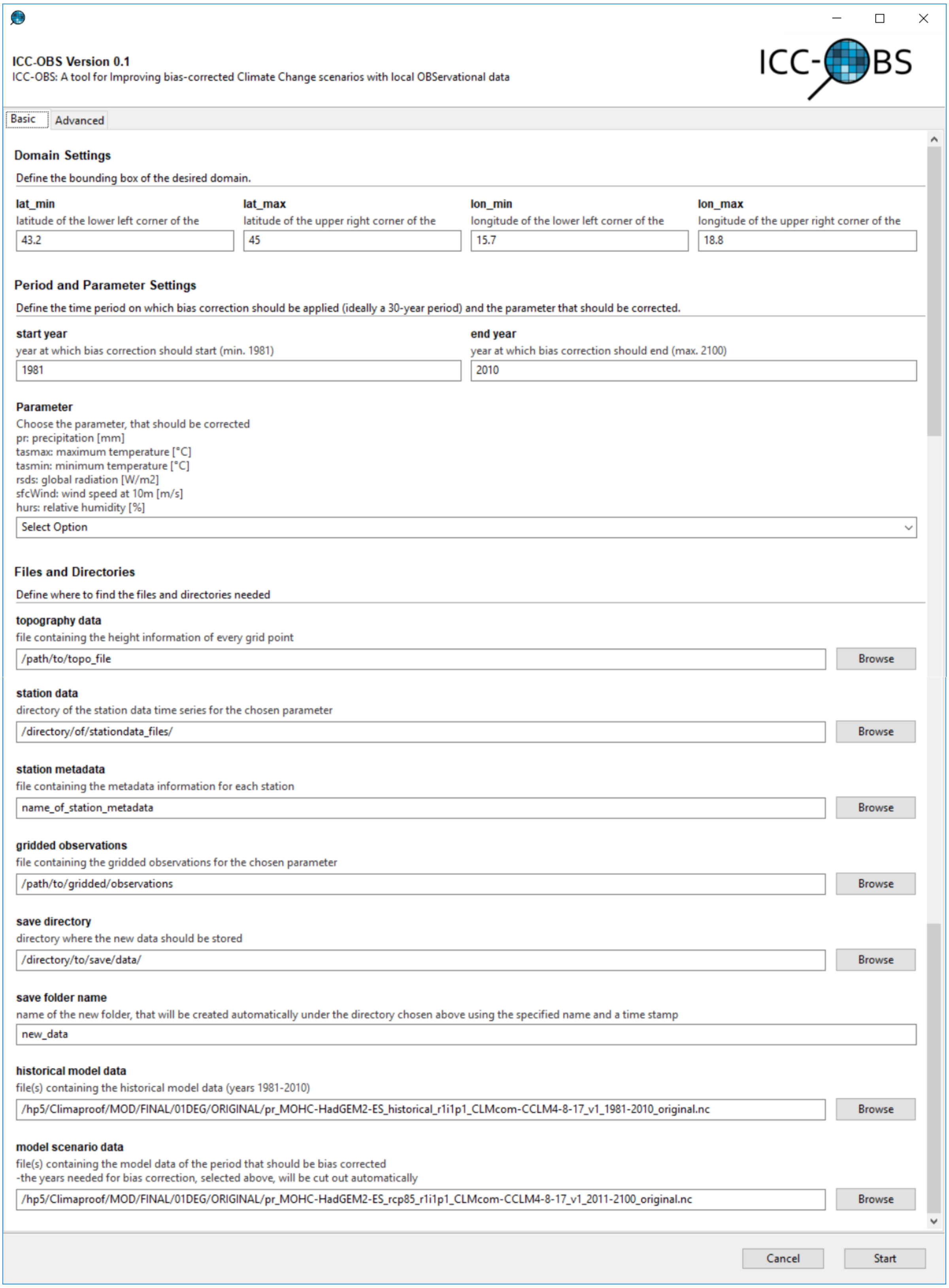

Figure 9: Screenshot of the ICC-OBS Tool graphical user interface.

1. Choose the Basic or Advanced Tab.

   If you choose the Basic Tab, default interpolation settings will be chosen (described below). In the Advanced tab you can modify the interpolation settings.

2. Define your domain.

   Specify the latitude/longitude values (in decimal degrees) of the lower left and upper right corner of your domain. This should cover the area, for which you have additional station data. If no station data is inside the defined domain, the tool will throw an error.

3. Define the bias-correction period

   Set the start and end year of the time period that should be bias corrected. The years can be chosen freely between 1981 and 2100. The correction period starts at the 1st of January of the defined start year and ends at the 31st of December of the defined end year.

   The choice of the bias-correction period does not affect the historical (calibration) period, which is always 1981-2010.

> The bias correction algorithm is implemented to use 30-year periods with a 10-year sliding window to correct the middle 10 years. Furthermore, the bias corrected data is stored in 10-year blocks. Hence, it is recommended to choose a time period of at least 30 years.

4. Choose the parameter for improved bias correction

   Select the parameter from the parameter list that should be bias corrected. This is also the parameter for which you have additional station data. Available parameters are listed in Table 1.

5. Select the topography file

   Select the file containing the height information for the common grid clicking on the "Browse" Button or typing in the path to the file manually. The grid resolution of the topography has to be the same as the resolution of the climate model and the gridded observation data.

6. Define the station data directory

   Define the directory (not a single file), that contains the station data time series for the parameter chosen above.

7. Define the station metadata file

   Select the file that contains the metadata for all stations in the format described above.

8. Select the file containing the gridded observations for the chosen parameter.

9. & 10. Define the save directory path and folder name

Define the path and name of the directory where the newly created data should be saved. The tool automatically creates a new directory under the defined path using the save folder name and a time stamp (*name_YYYY-MM-DD_hhmm*).

11. Select historical model data

Select the historical model of your choice for the parameter defined above. For help with the selection of the right model please refer to the ClimaProof Model Selection Tool.

12. Select model scenario data

Select the model scenario (RCP) of your choice corresponding with the historical model chosen above. If you want to correct only the historical period, the tool will automatically get the data for the correction period from the historical model data.

13. Press Start to run the tool

Interpolation settings

Regarding the interpolation of the station data to the model grid, you can either stick to the recommended default settings by staying in the "Basic" tab of the tool or modify the interpolation settings by selecting the "Advanced" tab (see Figure 10). There you can choose between ordinary kriging and inverse distance weighting interpolation (IDW).

Subsettings when choosing IDW:

- Radius of influence (in km): The radius from the grid center in km, within which observations are considered and weighted.
- Minimum neighbours: The minimum number of neighbours (stations) within the defined radius needed to perform IDW interpolation for that point.

Subsettings when choosing kriging:

- Variogram model: exponential, gaussian, linear or spherical

Default settings

Default settings for precipitation:

- Interpolation method: ordinary kriging (precipitation)
- Variogram model: gaussian

Default settings for all other parameters:

- Interpolation method: inverse distance weighting interpolation
- Radius of influence: average distance between the sampled points (stations)
- Minimum neighbours: 3

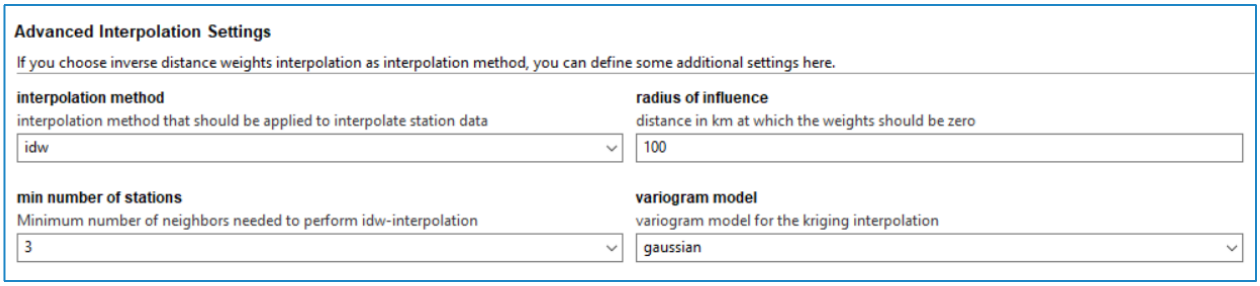

Figure 10: Screenshot of the section for the advanced interpolation settings in the “Advanced” tab.

Output Data

Once the ICC-OBS tool successfully finished, you can start exploring the output data. All the newly created data is stored in a separate folder created automatically by the tool using the "save folder name" you specified in the graphical user interface (GUI) and a time stamp (e.g. testdata_2019-04-03_1030). You can find the new folder under your specified save directory.

The new files and subdirectories that are created within the new folder are listed and described in Table 4. The italic parts of the filename are substituted by the respective parameters defined by the user via the GUI.

Table 4: Subdirectories and files created by the ICC-OBS Tool.

Subdirectory	**Filename**	**Description**
	*param*_merged_observations_1981-2010.nc	Improved observational data for the historical time period (1981-2010) and the specified parameter (pr, tasmax, tasmin, rsds, sfcWind, hurs)
TMP	*param*_original_gridded_obs.nc	Subset of the original gridded observations, cut out to the specified domain
TMP	*param*_model_historical_subset_1981-2010.nc	Subset of the original historical model data, cut out to the specified domain
TMP	param_model_bc_period_subset_*YYYY*.nc	Subset of the original model data for the defined bias correction period (start year - end year), cut out to the specified domain
BIASCORR_TS	*param*_y*_x*.nc	Grid point time series of bias corrected model data. For every grid point in the domain a separate file will be created.
BIASCORR	*param_model_YYYY*.nc	Improved bias corrected model data for the selected model and time period (saved in 10-year blocks)
PLOTS	*param_interpolationmethod*.png	Plot comparing the original and improved (with station data) observational data (mean over the historical period 1981-2010), that is created using the interpolation method defined in the GUI. Example: Figure 11
PLOTS	*param*_biascorr.png	Plot comparing the original (uncorrected) model data with the new bias corrected model data (mean over the bias correction period) Example: Figure 12

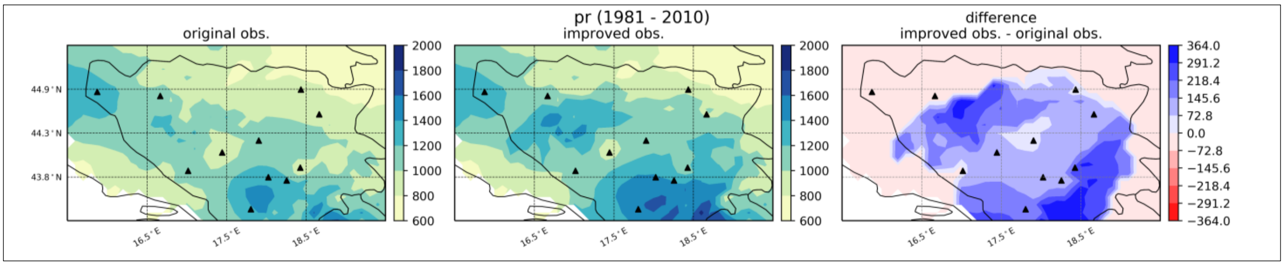

Figure 11: Example of an automatically created plot comparing original and improved gridded observations for yearly mean precipitation for the historical period 1981-2010.

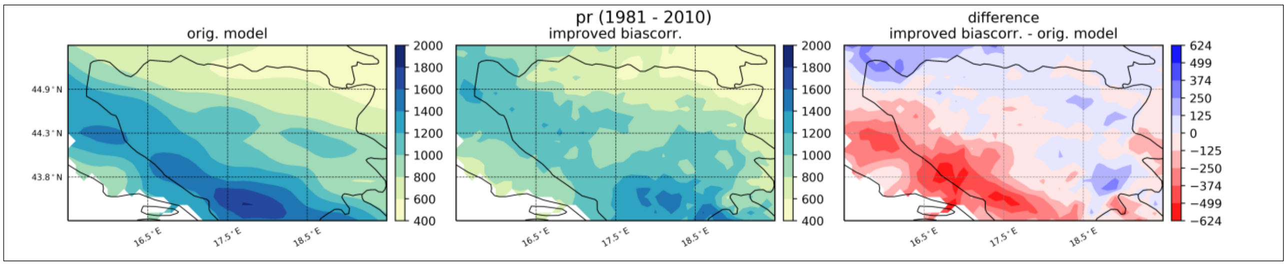

Figure 12: Example of an automatically created plot comparing original and bias corrected model data for yearly mean precipitation for the bias correction time period 1981-2010.

Notes

- Since the procedure is computationally expensive, the processed area should be as small as possible and only cover an area, where additional data is available.
- **The more stations, the better** the localization of the observational data. Especially when using kriging, a high number of stations is necessary for a good estimation of the variogram model and hence a reasonable interpolation.
- **Good quality of station data** is important. If the quality of the additional data is low, there won't be any improvement of the final dataset.
- Please check your data carefully for erroneous values before using it within the tool.
- The tool is still in it's beta phase and therefore errors can occur. If you find a bug, please report back to us and we will try to fix it.

The ClimaProof Downscaling Tool

For applications, that require a higher resolution than the 0.1° resolution that is available by default via the CCCA data server, the ClimaProof Downscaling Tool was developed. To run the tool, the following input data is required:

1. Gridded data at 0.1° resolution that should be downscaled (either model or observational data)
2. Coarse topography file (0.1°)
3. High resolution topography file (0.01°)

Attention: It is not advised to downscale climate indicators directly. Instead, download the desired climate parameters, downscale them with the downscaling tool and then use the indicator scripts on https://github.com/boku-met/climaproof-scripts to calculate the indicators with the resulting data.

The tool can only process one file at a time - if multiple files should be downscaled, you have to run the tool for each file separately.

A detailed description of the methods used within the tool is provided in the Methodical Background Document (https://github.com/boku-met/climaproof-docs)

Running the tool

Linux/MacOS

The installation process for the ClimaProof Downscaling Tool is analogous to the installation of the ClimaProof Model Selection Tool (the two tools are provided in one package). If you already installed the Model Selection Tool as described above, you can directly start the Downscaling Tool. If not, please refer to the installation guide of Installing the Python environment

The materials developed within the ClimaProof project are programmed in Python.

In order to run them on your local machine, you have to set up a Python Environment.

The easiest way to do this is via Miniconda - an open-source package management system that runs on Windows (64-bit), macOS and Linux. Furthermore, Miniconda allows you to install everything in your home-directory without the need for administrative or root permissions.

To set up the Python Environment please follow the step-by-step instruction below.

Notes on the installation:
Please be aware that the Python installation requires approximately 10GB of space on your disk so choose a directory that has enough space available.
Unfortunately, the Downscaling Tool currently only works under Linux and MacOS. As a workaround, Windows 10 users can run the tool through a Docker container. For this, Docker has to be installed on your Computer.

> **Attention (Miniconda users only)!**
>
> In the current versions of the ClimaProof tools (v1_3+), bugfixes in some of the required backend libraries were implemented. In order to update the backend libraries, writing permissions to the installed Miniconda folders by the user are required. The updates are done automatically the first time a tool is started.

Miniconda

Linux

1. Download Miniconda (Python 3) for Linux (https://conda.io/en/latest/miniconda.html):

   You can do that manually by clicking on the link above and downloading the installer with your browser or by opening a terminal and running:

   ```
   wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
   ```

2. Open a Terminal window, navigate to the directory containing the installation file and run:

   ```
   sh Miniconda3-latest-Linux-x86_64.sh
   ```

3. Follow the instructions on the installer screens - accept defaults settings (you can change them later).

4. Close and re-open your Terminal window to make the changes take effect.
   To test your installation, type:

```
conda list
```

   If your installation was successful, a list of installed packages appears.

5. Update Conda:

```
conda update conda
```

Windows

1. Download Miniconda (Python 3) for Windows (https://conda.io/en/latest/miniconda.html)
2. Start the .exe file and follow the instructions of the installer or
   open the Anaconda Prompt, navigate to the directory containing the installation file

```
cd directory
```

   e.g.

```
cd C:\Users\Me\Downloads
```

   and run:

```
start /wait "" Miniconda3-latest-Windows-x86_64.exe /InstallationType=JustMe /RegisterPython=0 /S
/D=%UserProfile%\Miniconda3
```

3. Open “Anaconda Promt” – a terminal window appears.
   To test your installation, type:

```
conda list
```

   If your installation was successful, a list of installed packages appears.

4. Update Conda

```
conda update conda
```

MacOS

1. The easiest way to install Miniconda on Mac is via homebrew (see homebrew website for current changes on how to install: https://brew.sh/).
   You can install homebrew by opening a terminal and entering:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

2. Once homebrew is installed, you can install Miniconda with

```
brew cask install miniconda
```

3. Close and re-open your Terminal window to make the changes take effect.
   To test your installation, type:

```
conda list
```

   If your installation was successful, a list of installed packages appears.

4. Update Conda:

```
conda update conda
```

Docker

The installation of Docker is only necessary if you want to use the ClimaProof Downscaling Tool under Windows. For the installation you need to have administrative permissions on your computer.

Docker Desktop for Windows runs on 64-bit Windows 10 Pro, Enterprise, and Education (1607 Anniversary update, Build 14393 or later). We currently cannot support older Windows versions.

Docker Desktop for Windows

Please follow the instructions on https://docs.docker.com/docker-for-windows/install to install Docker Desktop for Windows.

- Download Docker
- Install Docker
- Add user to Docker user group
- allow virtualization in BIOS settings

The Model Selection Tool.

Attention (Miniconda users only)!

In the current versions of the ClimaProof tools (v1_3+), bugfixes in some of the required backend libraries were implemented. In order to update the backend libraries, writing permissions to the installed Miniconda folders by the user are required. The updates are done automatically the first time a tool is started.

To start the tool, navigate to the *climaproof-tools* directory, activate the conda environment and start the tool:

```
cd climaproof-tools

conda activate desired_name_of_environment

bokeh serve --show dst
```

This will start the Downscaling Tool in a new tab in your browser.

Windows

If you want to run the Downscaling Tool under Windows, you have to install Docker first (see Page 17). Once Docker is installed and running, navigate to the *climaproof-tools* directory and run the following commands in the command line:

```
cd climaproof-tools

docker network create cproof

docker build --rm --network=cproof -t climaproof/tools

docker run -t -i -p 5100:5100 -v LOCAL_PATH_TO_DATA:/data climaproof/tools --network=cproof
```

The LOCAL_PATH_TO_DATA is your personal path to the data that you want to downscale (no subdirectories are allowed).

Then open your browser (e.g. Firefox) and go to the following address: http://127.0.0.1:5100/dst

Quitting the Tool (Miniconda users only)

Once you are finished with model selection, you can close the browser Tab. However, this does not terminate the Bokeh server running in the background. In your Terminal, Press CTRL + c to shut down the running server.

Downscaling tool - Step-by-Step guide

In your browser the Downscaling Tool will look like this:

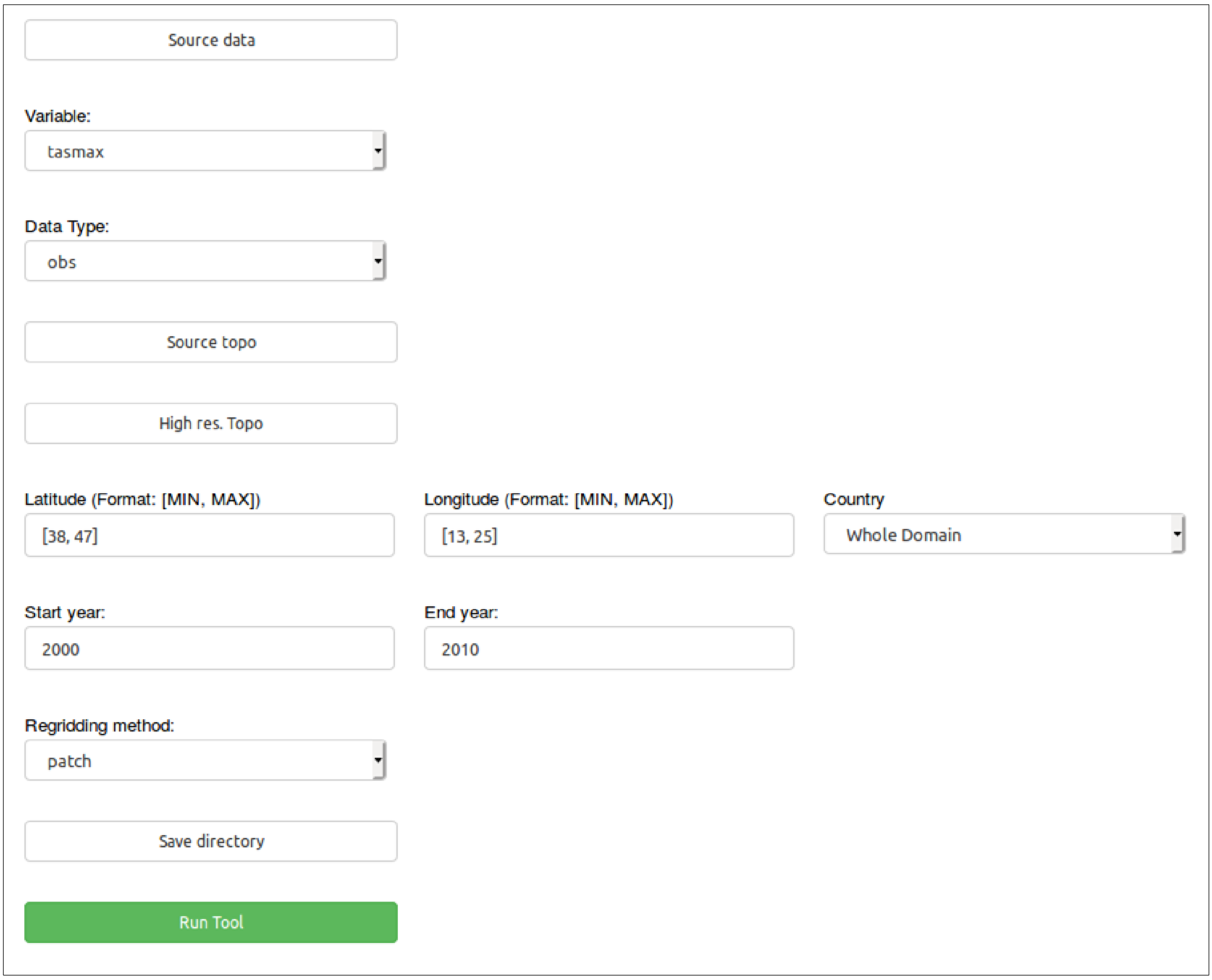

Figure 13: Screenshot of the Downscaling Tool.

To get the downscaled data, the following steps are needed:

1. Source data: Select the data file that should be downscaled to 0.01° resolution
2. Variable: Select the appropriate variable from the drop-down menu.
3. Data Type: Select whether the source data is observational (obs) or climate model data (model)
4. Source topo: Select the topography file that corresponds to the source data (0.1°). This data can be downloaded via the CCCA Data Centre (Select filter 'Variables' -> 'height above surface')
5. High resolution topo: Select the file that contains the high-resolution (0.01°) topography information. This data can be downloaded via the CCCA Data Centre (Select filter 'Variables' -> 'height above surface')
6. Select the bounding box of the desired domain. Type in latitude and longitude (in decimal degrees) or select a country. Latitude and longitude are automatically updated if you choose a country from the drop-down menu.
7. Define the time frame for which the downscaling should be done by typing in start- and end-year (the end year is always included in the computation).
8. Choose between the regridding methods "patch" and "bilinear" (more information on the methods can be found in the Methodical Background document or under https://xesmf.readthedocs.io/en/latest/notebooks/Compare_algorithms.html).
9. Save directory: Define the directory where the final data should be stored.
10. Press the "Run Tool" button to start the computation.

Note: If you encounter an error message similar to this:

```
------------ ERROR ------------
('Regrid(filename) requires PIO and does not work if ESMF has not been built with MPI support',)
```

The reason could be that the Python Package *xesmf* that contains the regridding algorithms has not been built with *MPI* support. To solve this problem, uninstall *xesmf* with

```
conda uninstall xesmf
```

then install *MPI* with

```
conda install mpi
```

Next, re-install *xesmf* with

```
conda install -c conda-forge xesmf
```

Attention! Do NOT update this conda environment. The tools have been written and tested for the specific package versions of this environment. Updating it to the latest version will stop the tools from working correctly.

The algorithm is computationally very expensive. Older machines or machines with small memory can run into memory errors. If this happens, please try to select a smaller domain or a shorter time frame.

Output

Once the computation is finished, visualization is created automatically below the input fields (not available when running the downscaling tool on Windows with Docker). The plot compares the coarse data (at 0.1° resolution) with the newly created downscaled data (at 0.01° resolution). You can furthermore choose the season you want to display and zoom into a specific area you are interested.

Additionally, the output data is stored under the specified save directory as a netCDF file. The naming convention for the data is: *variable_datatype_startyear-endyear.nc* (e.g. *tasmax_observations_1981-1990.nc*)

Calculate climate indicators with custom data

The calculation routines for the climate indicators are provided on GitHub:

https://github.com/boku-met/climaproof-scripts

You can use the scripts to calculate the ClimaProof climate indicators for your own customized data.

The user guide for the scripts is also available in the readme file of the repository.

Welcome!

In this repository (https://github.com/boku-met/climaproof-scripts) you can find the scripts used to calculate climate indicators in the ClimaProof project. The programming language used is Python. In order to run the scripts and calculate indicators, you need to follow a few steps that are listed in this file.

All climate indicators have been published for all available climate scenarios of the entire Western Balkan region:

https://data.ccca.ac.at/group/ec12144b-a8f1-40bc-9297-931bcfd01b5a?groups=ci

So there is only reason to use these scripts if you have customised the climate scenarios by improving data resolution (downscaling tool) and/or data quality with observations (ICC-OBS tool).

Step 1: Download the scripts

If you already know which indicator you want to calculate, you have several options of downloading the scripts:

- Click on the name of the script you are interested in and copy the code into an empty text file. You can then save the file under "name_of_the_script.py"
- Download all scripts by clicking on the green "Code" icon, and then "Download ZIP"
- If you have git installed on your local machine, you can clone this repository with

```
git clone https://github.com/boku-met/climaproof-scripts.git
```

Step 2: Setting up the environment

The scipts are developed to run under the same conda environment as the ClimaProof tools. So if you already set up the environment described in chapter Installing the Python environment you are good to go. If not, please do so.

Step 3: Get the data

Each climate indicator requires certain climate parameters. Please make sure to check the scripts to see which parameters are required:

Open script - look for text block "def user_data()" at the top - first comment specifies the required input data

You can use the ClimaProof tools to customise the available climate scenarios or download them directly: https://data.ccca.ac.at/group/climaproof

The scripts use only one file path to look for input data, so make sure to put all data into the same folder.

Step 4: Specify file paths and user info

Each script contains a section labeled "def user_data()" at the beginning. In this section, you can specify the path to the required input data, and a few additional options. The file paths need to be written within quotation marks (""). Each user-defined setting is explained by a comment. When you finished setting the user details, make sure to save the script text file, otherwise it will not work.

Attention:

Starting the scripts without setting the user info will not work. Starting them with the wrong info might lead to unwanted results. Always check that the user info is correct before you run the scripts.

Step 5: Run the scrips

When you have the conda environment installed (Step 2), running the scripts is straightforward. In the command line, type

```
conda activate tools (or user-defined name of the ClimaProof environment)

python name_of_the_script.py
```

The scripts produce some text output. If you want to avoid the text in your command line, you can direct the output to a text file:

```
python name_of_the_script.py > name_of_textfile.txt
```

The scripts can take some time if they need to process large files. You can also run the programme in the background (Linux and MacOS only):

```
python name_of_the_script.py &

python name_of_the_script.py > name_of_textfile.txt &
```

For a hint how to run the scripts in the background under Windows, see https://superuser.com/questions/198525/how-can-i-execute-a-windows-command-line-in-background

If you have any questions or run into troubles with the scripts, please contact us!

Use cases

This chapter details an example that shows how all the tools, data and materials that were developed in the ClimaProof project could be combined in one comprehensive workflow. Further use cases are listed at the end of the chapter.

Climate-proofing a local infrastructure project

Assumption:

A new railroad section is proposed, the route planning should take future climate risks into account.

Basis for decision:

High-resolution climate indicators for critical risks to railroads.

Steps:

1) Select relevant indicators from Table 3

2) Note the required climate parameters

3) Select models with extreme values per scenario from the model selection tool (chapter The Model Selection Tool)
   - For the project region
   - Planning horizon: 2050

4) Download the data from the CCCA data centre (https://data.ccca.ac.at/group/climaproof)
   - Parameters necessary for indicator calculation
   - Models noted from the model selection tool

5a) Improve the climate scenarios with better observations (chapter The ICC-OBS Tool)
   - High quality station data available in the project region?

- Download original, non-bias corrected climate scenarios from the CCCA Data Centre (https://data.ccca.ac.at/group/ec12144b-a8f1-40bc-9297-931bcfd01b5a?text=regridded&groups=model_data)
- Use ICC-OBS tool!

5b) Continue with bias corrected climate scenarios

6) Downscale the climate parameters (chapter The ClimaProof Downscaling Tool)
   - Use downscaling tool
   - Only for the project region (computationally expensive!)

7) Calculate high-resolution climate indicators for the project region
   - All calculation scripts are available on https://github.com/boku-met/climaproof-scripts
   - Resulting files include self-describing metadata

Further use cases

- Large-scale climate impact assessment (coarse climate scenario and indicator data)
- Comparison of ensemble median with single models (for climate indicators/impact studies)
- Climate impacts and adaptation planning for other sectors
- High-resolution input data for impact models
- …

Annex

Table 5: List of EURO-CORDEX models compiling the bias-corrected ensemble

EURO-CORDEX Models	Resolution
CNRM-CERFACS-CNRM-CM5_rcp26_r1i1p1_CNRM-ALADIN53_v1	0.11°
CNRM-CERFACS-CNRM-CM5_rcp45_r1i1p1_CNRM-ALADIN53_v1	0.11°
CNRM-CERFACS-CNRM-CM5_rcp85_r1i1p1_CNRM-ALADIN53_v1	0.11°
CNRM-CERFACS-CNRM-CM5_rcp45_r1i1p1_CLMcom-CCLM4-8-17_v1	0.11°
CNRM-CERFACS-CNRM-CM5_rcp85_r1i1p1_CLMcom-CCLM4-8-17_v1	0.11°
CNRM-CERFACS-CNRM-CM5_rcp45_r1i1p1_SMHI-RCA4_v1	0.11°
CNRM-CERFACS-CNRM-CM5_rcp85_r1i1p1_SMHI-RCA4_v1	0.11°
ICHEC-EC-EARTH_rcp26_r12i1p1_CLMcom-CCLM4-8-17_v1	0.11°
ICHEC-EC-EARTH_rcp45_r12i1p1_CLMcom-CCLM4-8-17_v1	0.11°
ICHEC-EC-EARTH_rcp85_r12i1p1_CLMcom-CCLM4-8-17_v1	0.11°
ICHEC-EC-EARTH_rcp26_r12i1p1_KNMI-RACMO22E_v1	0.11°
ICHEC-EC-EARTH_rcp45_r12i1p1_KNMI-RACMO22E_v1	0.11°
ICHEC-EC-EARTH_rcp85_r12i1p1_KNMI-RACMO22E_v1	0.11°
ICHEC-EC-EARTH_rcp45_r1i1p1_KNMI-RACMO22E_v1	0.11°
ICHEC-EC-EARTH_rcp85_r1i1p1_KNMI-RACMO22E_v1	0.11°
ICHEC-EC-EARTH_rcp26_r3i1p1_DMI-HIRHAM5_v1	0.11°
ICHEC-EC-EARTH_rcp45_r3i1p1_DMI-HIRHAM5_v1	0.11°
ICHEC-EC-EARTH_rcp85_r3i1p1_DMI-HIRHAM5_v1	0.11°
IPSL-IPSL-CM5A-MR_rcp45_r1i1p1_IPSL-INERIS-WRF331F_v1	0.11°
IPSL-IPSL-CM5A-MR_rcp85_r1i1p1_IPSL-INERIS-WRF331F_v1	0.11°
IPSL-IPSL-CM5A-MR_rcp45_r1i1p1_SMHI-RCA4_v1	0.11°
IPSL-IPSL-CM5A-MR_rcp85_r1i1p1_SMHI-RCA4_v1	0.11°
MOHC-HadGEM2-ES_rcp26_r1i1p1_SMHI-RCA4_v1	0.11°
MOHC-HadGEM2-ES_rcp45_r1i1p1_SMHI-RCA4_v1	0.11°
MOHC-HadGEM2-ES_rcp85_r1i1p1_SMHI-RCA4_v1	0.11°
MOHC-HadGEM2-ES_rcp45_r1i1p1_CLMcom-CCLM4-8-17_v1	0.11°
MOHC-HadGEM2-ES_rcp85_r1i1p1_CLMcom-CCLM4-8-17_v1	0.11°
MOHC-HadGEM2-ES_rcp45_r1i1p1_KNMI-RACMO22E_v2	0.11°
MOHC-HadGEM2-ES_rcp85_r1i1p1_KNMI-RACMO22E_v2	0.11°
MPI-M-MPI-ESM-LR_rcp45_r1i1p1_CLMcom-CCLM4-8-17_v1	0.11°
MPI-M-MPI-ESM-LR_rcp85_r1i1p1_CLMcom-CCLM4-8-17_v1	0.11°
MPI-M-MPI-ESM-LR_rcp45_r1i1p1_MPI-CSC-REMO2009_v1	0.11°
MPI-M-MPI-ESM-LR_rcp85_r1i1p1_MPI-CSC-REMO2009_v1	0.11°
MPI-M-MPI-ESM-LR_rcp45_r2i1p1_MPI-CSC-REMO2009_v1	0.11°
MPI-M-MPI-ESM-LR_rcp85_r2i1p1_MPI-CSC-REMO2009_v1	0.11°
MPI-M-MPI-ESM-LR_rcp26_r1i1p1_SMHI-RCA4_v1a	0.11°
MPI-M-MPI-ESM-LR_rcp45_r1i1p1_SMHI-RCA4_v1a	0.11°
MPI-M-MPI-ESM-LR_rcp85_r1i1p1_SMHI-RCA4_v1a	0.11°
NCC-NorESM1-M_rcp45_r1i1p1_DMI-HIRHAM5_v2	0.11°
NCC-NorESM1-M_rcp85_r1i1p1_DMI-HIRHAM5_v2	0.11°

Table 6: List of MED-CORDEX models compiling the bias-corrected ensemble

MEDCORDEX Model (Atmosphere RCM)	**Resolution**
MED-11_ICTP-RegCM4_rcp85_r1i1p1_ICTP-RegCM4-3_v1	0.11°
MED-11_CNRM-CM5_rcp45_r8i1p1_CNRM-ALADIN52_v1	0.11°
MED-11_CNRM-CM5_rcp85_r8i1p1_CNRM-ALADIN52_v1	0.11°
MEDCORDEX Model (Fully-Coupled RCM)	
MED-44i_MPI-ESM-LR_rcp85_r1i1p1_UNIBELGRADE-EBUPOM2c_v1	0.44°