Better Tailoring of Climate Information for End Users using Targeted Interfaces and Tools

Christian Pagé
CERFACS, Toulouse, France

Abel Aoun
CERFACS Toulouse, France

Alessandro Spinuso
KNMI, De Bilt, Netherlands

Lars Bärring, Klaus Zimmermann
SMHI, Norrköping, Sweden

EGU General Assembly 2022
Vienna, Austria
May 23-27, 2022

IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

Climate Data Distribution through the ESGF

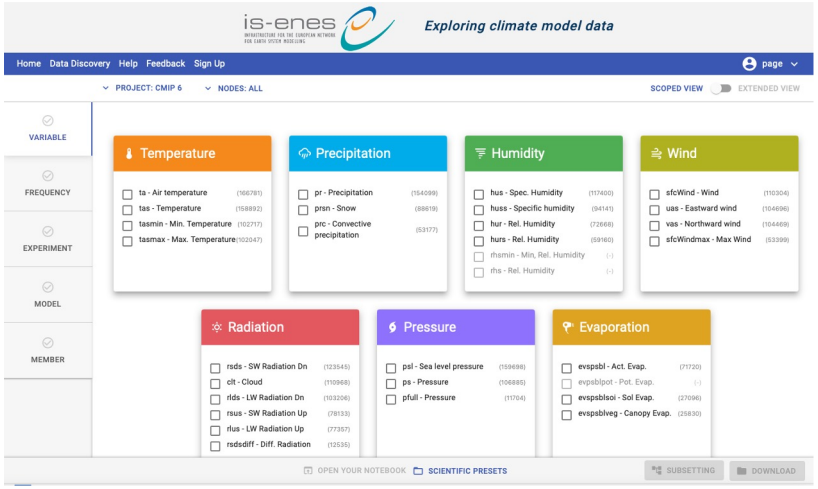
ESGF represents a multinational effort to securely **access**, **monitor**, **catalog**, **transport**, and **distribute** reference **data** for **climate** research experiments and observations.

CERFACS IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

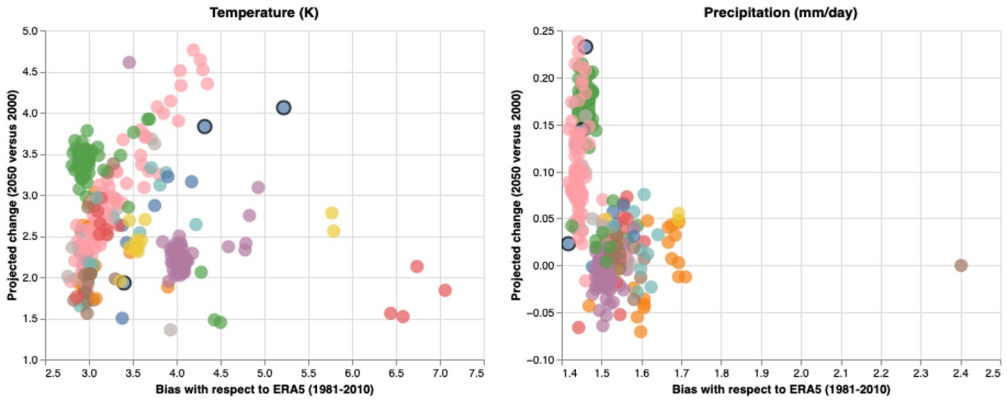
What is the climate4impact portal?

- Platform for researchers to explore climate data and perform analysis
- Front-end to climate data infrastructure
  - CMIP6, CMIP5, CORDEX, ...
- Jupyter-Lab enhanced environment (with SWIRRL) and Notebook gallery
- Climate indices calculations (**icclim**)
  - Personal store for processing outcomes
- Beta version available at https://dev.climate4impact.eu

Tailored Search Interface



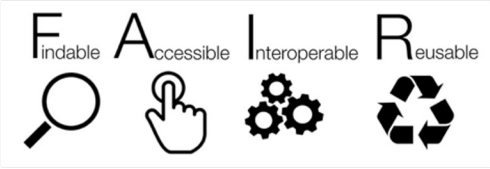
Climate Model performance comparison (**ESMValTool**)



IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

V2: Complete Redesign from current V1

- **GUI usability & Help/Feedback pages**
- **Flexible analysis features** (Notebooks with ICCLIM - Data Staging/Reduction Workflows)
- **Automated reproducibility mechanisms and documentation** (Data/Analysis)

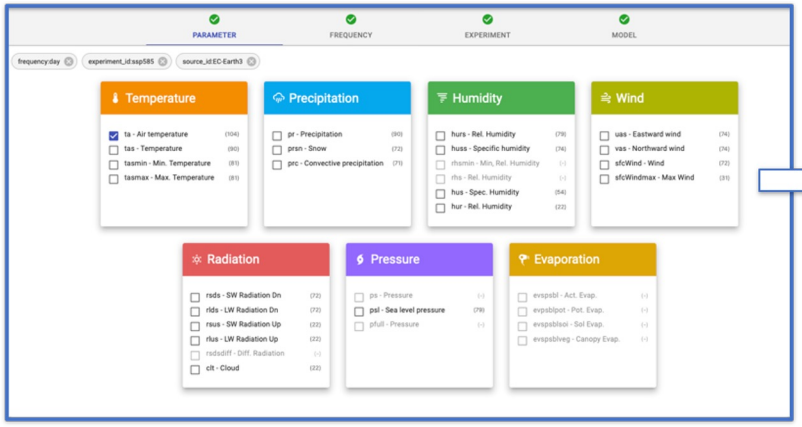


- **Pages for Models Performance Comparison** (ESMValTool)
- **Modular Deployment & Decoupled Architecture**

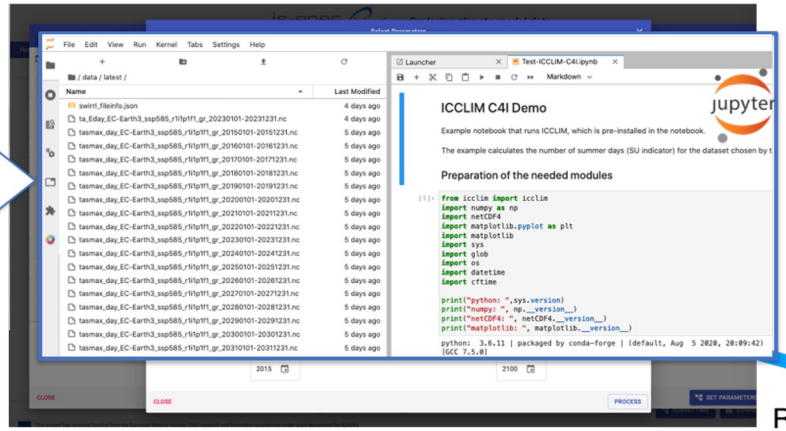
IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

Climate4Impact (v2) Workflows & Workspaces

Climate4Impact Search for CMIP5/6 CORDEX Data (Distributed Data)
https://dev.climate4impact.eu



Workflows for data staging & remote subsetting-reduction (WPS) onto Customisable Notebooks

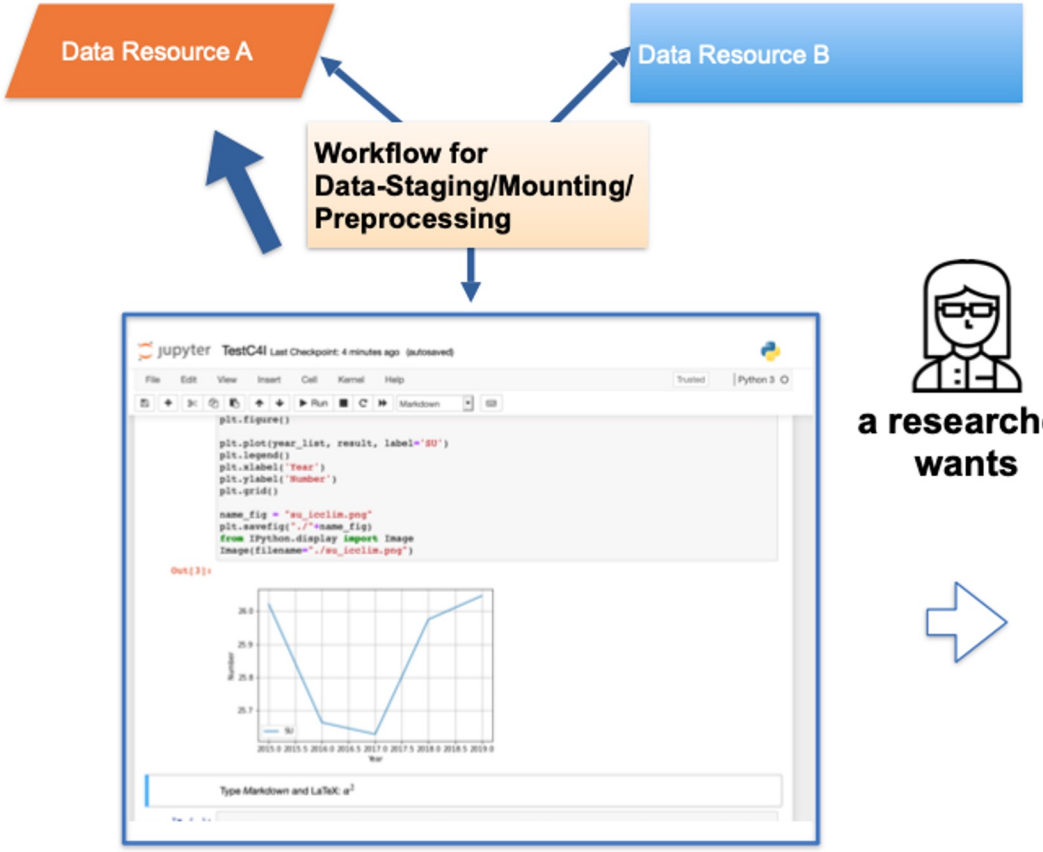


- Trace Changes to Restore, Recover Software and/or Data

W3C PROV

CERFACS IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

C4I Workspace Use Case

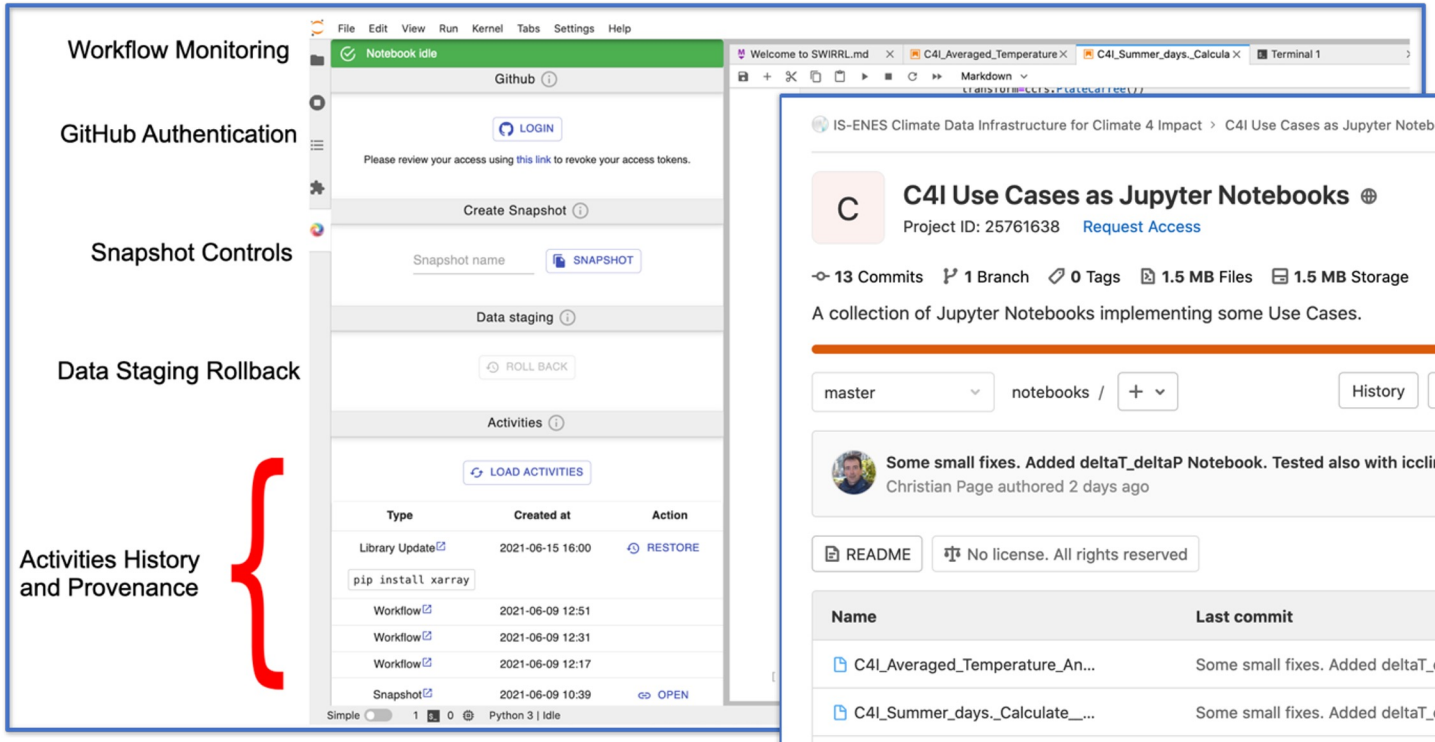


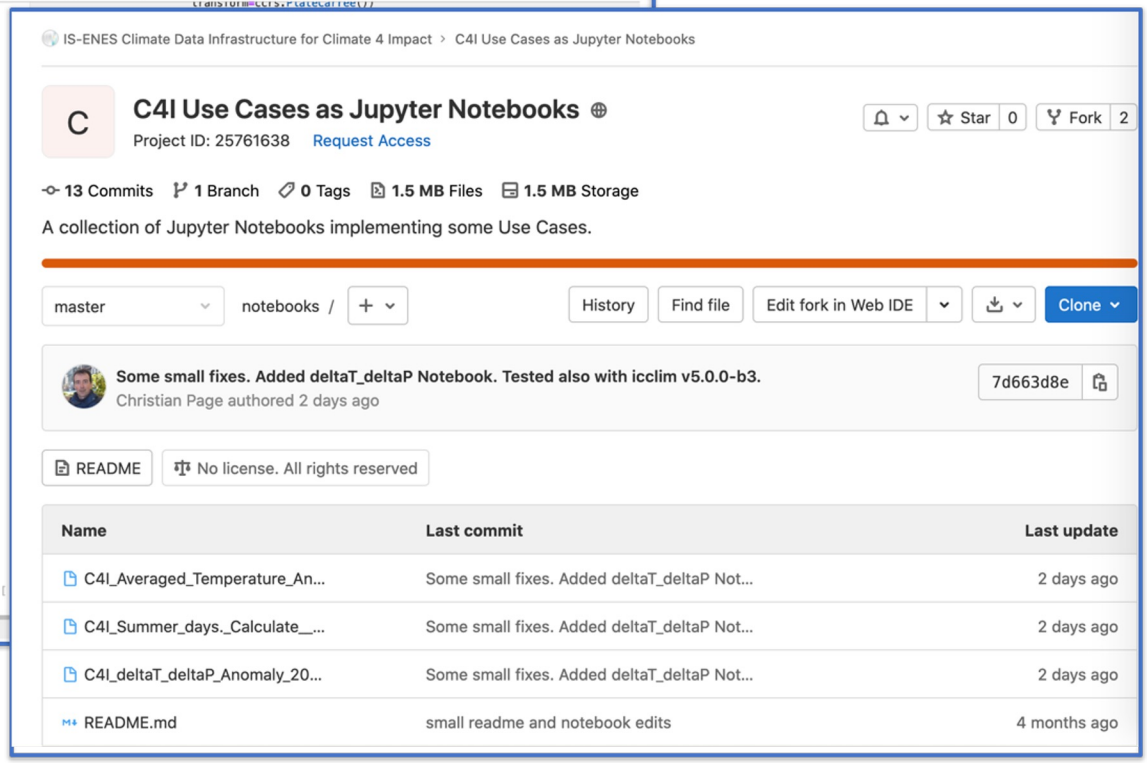
a researcher wants

- **access distributed raw data**
- **develop, document and reuse** methods for processing and visualisation.
- **update/extend** raw data and software
- **Track changes and rollback** (Traceability/Recovery)
- **keep old versions of the data** after updates (Reproducibility)
- **snapshot and restore** the state of a workspace software (Reproducibility)

CERFACS IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

SWIRRL JupyterLab Extension and Sample Notebooks based on icclim (climate indices)



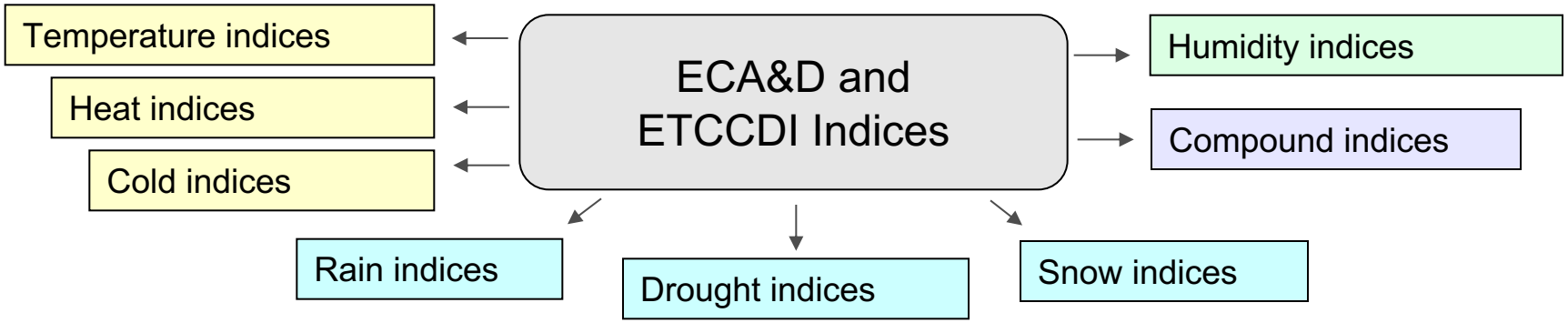


https://gitlab.com/is-enes-cdi-c4i/notebooks

IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

On-demand calculations

Climate indices calculation in climate4impact: **icclim**



- Intra-period extreme temperature range [°C] - **ETR**
- Warm days (days with mean temperature > 90th percentile of daily mean temperature) - **TG90p**
- Summer days (days with max temperature > 25 °C) - **SU**
- ...

- Python code developed at Cerfacs since September 2013
- Funded by EU FP7 IS-ENES2, FP7 CLIPC and H2020 IS-ENES3
- Generic and modular approach, can be reused in other environments
- New V5 completely rewritten and using underlying xclim functions, based on xarray and dask
- I/O interface is structured for optimal performance
- Implement the proper percentile indices calculations when calculation period overlaps reference period (called bootstrapping method)

CERFACS IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

icclim: climate indices

Documentation: https://icclim.readthedocs.io/en/latest/python_api.html
Source code: https://github.com/cerfacs-globc/icclim
Current Version 5.2.1: https://github.com/cerfacs-globc/icclim/releases/tag/5.2.1

`icclim.index(**kwargs)`

Parameters:

- **in_files** (*str | list[str] | Dataset | DataArray,*) – Absolute path(s) to NetCDF dataset(s), including OPeNDAP URLs, or path to zarr store, or xarray.Dataset or xarray.DataArray.
- **index_name** (*str*) – Climate index name. For ECA&D index, case insensitive name used to lookup the index. For user index, it's the name of the output variable.
- **var_name** (*str | list[str] | None*) – `optional` Target variable name to process corresponding to `in_files`. If None (default) on ECA&D index, the variable is guessed based on the climate index wanted. Mandatory for a user index.
- **slice_mode** (*str*) – Type of temporal aggregation: {"year", "month", "DJF", "MAM", "JJA", "SON", "ONDJFM" or "AMJJAS"}. Default is "year". See slice_mode for details.
- **time_range** (*list[datetime.datetime]*) – `optional` Temporal range: upper and lower bounds for temporal subsetting. If `None`, whole period of input files will be processed. Default is `None`.
- **out_file** (*str | None*) – Output NetCDF file name (default: "icclim_out.nc" in the current directory). Default is "icclim_out.nc". If the input `in_files` is a `Dataset`, `out_file` field is ignored. Use the function returned value instead to retrieve the computed value. If `out_file` already exists, icclim will overwrite it!
- **threshold** (*float | list[float] | None*) – `optional` User defined threshold for certain indices. Default depend on the index, see their individual definition. When a list of threshold is provided, the index will be computed for each thresholds.

CERFACS IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

icclim: climate indices

Correspondence table "index - source variable"

Using common names for the source variable, icclim is able to lookup the proper variable in the given input to compute an index.

index	Source variable
TG, GD4, HD17, TG10p, TG90p	daily mean temperature
TN, TNx, TNn, TR, FD, CFD, TN10p, TN90p, CSDI	daily minimum temperature
TX, TXx, TXn, SU, CSU, ID, TX10p, TX90p, WSDI	daily maximum temperature
DTR, ETR, vDTR	daily maximum + daily minimum temperature
PRCPTOT, RR1, SDII, CWD, CDD, R10mm, R20mm, RX1day, RX5day, R75p, R75pTOT, R95p, R95pTOT, R99p, R99pTOT	daily precipitation flux (liquide phase)
SD, SD1, SD5cm, SD50cm	daily snowfall flux (solid phase)
CD, CW, WD, WW	daily mean temperature + daily precipitation flux (liquide phase)

IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

ICCLIM C4I: Calculate the percentage of days when Tmax > 90th percentil (TX90p)

Example notebook that runs ICCLIM, which is pre-installed in the notebook.

The example calculates the percentage of days when Tmax > 90th percenti

The data is read using xarray and a plot of the time series over a specific re
examples are shown.

The dataset that is expected for this notebook are tasmax parameter (need
experiment as well as one member. The time period should be continuous.

The following time period is considered: 2081-01-01 to 2100-12-31 using th
European region.

https://gitlab.com/is-enes-cdi-c4i/notebooks

icclim: https://icclim.readthedocs.io/

Preparation of the needed modules

[1]:
```
import icclim

import sys
import glob
import os
import datetime
import cftime

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import cartopy.crs as ccrs

print("python: ",sys.version)
print("numpy: ", np.__version__)
print("pandas: ", pd.__version__)
```

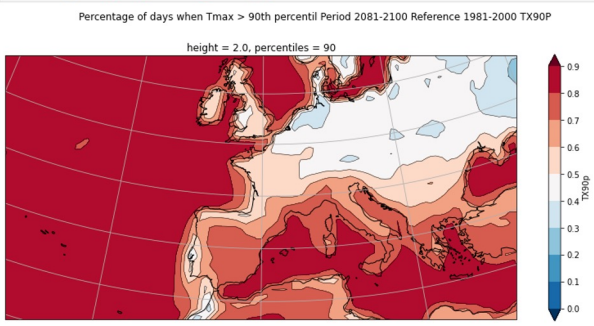
Specification of the parameters and period of interest

[2]:
```
# studied period
dt1 = datetime.datetime(2081,1,1)
dt2 = datetime.datetime(2100,12,31)

# reference period
dt1r = datetime.datetime(1981,1,1)
dt2r = datetime.datetime(2000,12,31)

out_f = 'tx90p_icclim.nc'
filenames = glob.glob('./data/latest/tasmax_day*.nc')

icclim.indice(indice_name='TX90p', in_files=filenames, var_name='tasmax', slice_mode='JJA', base_period_time_range=[dt1r, dt2r], t
```

```
2021-10-15 07:34:45,151    ********************************************************************************
2021-10-15 07:34:45,151    *                                                                              *
2021-10-15 07:34:45,152    *          Icclim               V5.0.0b5      *
2021-10-15 07:34:45,153    *                                                                              *
2021-10-15 07:34:45,153    *                                                                              *
2021-10-15 07:34:45,154    *          Fri Oct 15 07:34:45 2021 GMT                                        *
2021-10-15 07:34:45,154    *                                                                              *
2021-10-15 07:34:45,155    *          BEGIN EXECUTION                                                     *
2021-10-15 07:34:45,155    *                                                                              *
2021-10-15 07:34:45,156    ********************************************************************************
2021-10-15 07:34:45,156 Processing: 0%
```

```
p = tx90_avg.plot.contour(levels=levels,
                          colors='k',
                          linewidths=0.5,
                          transform=ccrs.PlateCarree())

# Contour filled colors
p = tx90_avg.plot.contourf(levels=levels,
                           cmap='RdBu_r',
                           extend='both',
                           transform=ccrs.PlateCarree())

# Plot information
plt.suptitle("Percentage of days when Tmax > 90th percentil Period 2081-2100 Reference 1981-2000 TX90P", y=1)

# Add the coastlines to axis and set extent
ax.coastlines()
ax.gridlines()
ax.set_extent(extent)

# Save plot as png
plt.savefig('c4i_tx90p_contours_icclim.png')
```

```
lim/core/cfchecks.py:39: UserWarning: Variable has a non-confor
: maximum*']`
```



CERFACS IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

Thanks !

On behalf of the climate4impact and icclim teams

Christian Pagé christian.page@cerfacs.fr
Alessandro Spinuso alessandro.spinuso@knmi.nl
Abel Aoun abel.aoun@cerfacs.fr

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N°824084

Our website
https://is.enes.org/

Follow us on Twitter !
@ISENES_RI

Contact us at
is-enes@ipsl.fr

Follow our channel
IS-ENES3 H2020

THE CONSORTIUM

Coordinated by **CNRS-IPSL**, the IS-ENES3 project gathers **22 partners** in **11 countries**

IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084