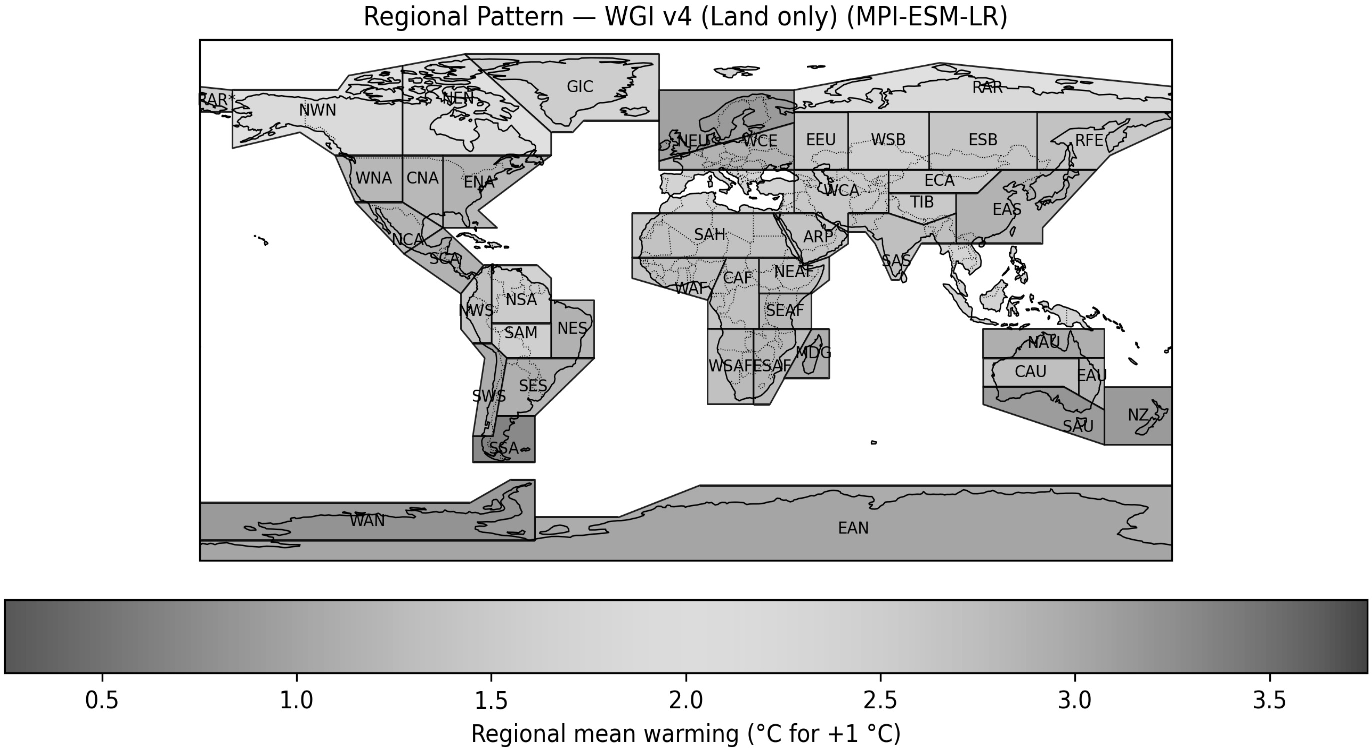
Building Interpretable Climate Emulators for Economics



Description of programs and datasets used

Organization of the repository

This Python-based code repository supplements the work of Aryan Eftekhari, Doris Folini, Aleksandra Friedl, Felix Kuebler, Simon Scheidegger, and Olaf Schenk titled *Building Interpretable Climate Emulators for Economics* (Eftekhari et al.; 2025).

- This repository contains three distinct folders:
  1. "data": Raw data for *Section 3 - A Calibrated Climate Emulator*, and *Appendix A, B* of our article.
     - Its content and usage are detailed in the corresponding README.
  2. "DEQN": Replication codes for *Section 4 - Numerical Experiments*, where non-stationary integrated assessment models (IAMs) are solved by adopting "Deep Equilibrium Nets (DEQN)" to the context of climate economic models.
     - How to reproduce the model solutions from scratch as well as the content and postprocessing of the pretrained solutions is detailed in the readme that provided under the following link: README.
  3. "figures_replication": Replication routines for plotting all the figures that are presented in the paper.
     - Its content and usage are detailed in the corresponding README.

Replication of the numerical results

- To replicate the results of the article step-by-step, a detailed set of instructions is provided *here*.

Datasets used

- The detailed references to the datasets used in our work are provided *here*.

Statement about Rights

We certify that the authors of the manuscript have legitimate access to and permission to use the data used in this manuscript.

Summary of Availability

All data **are** publicly available. The description and the links to the datasets can be found *here*.

Datasets

We use various datasets as input to our computations, all of which are briefly explained below. Furthermore, we provide the relevant URLs, and include them in this repository for convenience. The relevant references to the datasets we are using are

Meinshausen et al. (2011), Taylor et al. (2012), Gasser et al. (2020), Joos et al. (2013), Jones et al. (2019), Macdougall et al. (2020), Lynch et al. (2017), Kravitz et al. (2017), Iturbide et al. (2020):

```
@article{taylor-et-al:12,
    author = {{Taylor}, Karl~E. and {Stouffer}, Ronald~J. and {Meehl}, Gerald~A.},
    title = "{An Overview of CMIP5 and the Experiment Design}",
    journal = {Bull. Amer. Meteor. Soc.},
    year = 2012,
    volume = 93,
    pages = {485-498},
    doi = {10.1175/BAMS-D-11-00094.1}
}

@article{meinshausen-et-al:11,
    author = {{Meinshausen}, Malte and {Smith}, S.~J. and {Calvin}, K. and {Daniel}, J.~S. and {Kainuma}
    title = "{The RCP greenhouse gas concentrations and their extensions from 1765 to 2300}",
    journal = {Climatic Change},
    year = 2011,
    month = nov,
    volume = {109},
    number = {1-2},
    pages = {213-241},
    doi = {10.1007/s10584-011-0156-z},
    adsurl = {https://ui.adsabs.harvard.edu/abs/2011ClCh..109..213M},
    adsnote = {Provided by the SAO/NASA Astrophysics Data System}
}

@article{gasser2020historical,
    title={Historical CO 2 emissions from land use and land cover change and their uncertainty},
    author={Gasser, Thomas and Crepin, L{\'e}a and Quilcaille, Yann and Houghton, Richard A and Ciais, P
    journal={Biogeosciences},
    volume={17},
    number={15},
    pages={4075--4101},
    year={2020},
    publisher={Copernicus Publications G{\"o}ttingen, Germany}
}

@article{joos2013carbon,
    title={Carbon dioxide and climate impulse response functions for the computation of greenhouse gas m
    author={Joos, Fortunat and Roth, Raphael and Fuglestvedt, Jan S and Peters, Glen P and Enting, Ian G
    journal={Atmospheric Chemistry and Physics},
    volume={13},
    number={5},
    pages={2793--2825},
    year={2013},
    publisher={Copernicus GmbH}
}

@article{jones2019zero,
    title={The Zero Emissions Commitment Model Intercomparison Project (ZECMIP) contribution to C4MIP: q
    author={Jones, Chris D and Fr{\"o}licher, Thomas L and Koven, Charles and MacDougall, Andrew H and M
    journal={Geoscientific Model Development},
    volume={12},
    number={10},
    pages={4375--4385},
    year={2019},
    publisher={Copernicus GmbH}
}

@article{macdougall2020there,
    title={Is there warming in the pipeline? A multi-model analysis of the Zero Emissions Commitment fro
    author={MacDougall, Andrew H and Fr{\"o}licher, Thomas L and Jones, Chris D and Rogelj, Joeri and Ma
    journal={Biogeosciences},
    volume={17},
    number={11},
    pages={2987--3016},
    year={2020},
```

```
    publisher={Copernicus Publications G{\"o}ttingen, Germany}
}

@article{lynch2017open,
  title={An open-access CMIP5 pattern library for temperature and precipitation: description and methodo
  author={Lynch, Cary and Hartin, Corinne and Bond-Lamberty, Ben and Kravitz, Ben},
  journal={Earth System Science Data},
  volume={9},
  number={1},
  pages={281--292},
  year={2017},
  publisher={Copernicus GmbH}
}

@article{kravitz2017exploring,
  title={Exploring precipitation pattern scaling methodologies and robustness among CMIP5 models},
  author={Kravitz, Ben and Lynch, Cary and Hartin, Corinne and Bond-Lamberty, Ben},
  journal={Geoscientific Model Development},
  volume={10},
  number={5},
  pages={1889--1902},
  year={2017},
  publisher={Copernicus GmbH}
}

@article{iturbide2020update,
  title={An update of IPCC climate reference regions for subcontinental analysis of climate model data:
  author={Iturbide, Maialen and Guti{\'e}rrez, Jos{\'e} M and Alves, Lincoln M and Bedia, Joaqu{\'\i}n a
  journal={Earth System Science Data},
  volume={12},
  number={4},
  pages={2959--2970},
  year={2020},
  publisher={Copernicus Publications G{\"o}ttingen, Germany}
}
```

CMIP5 output data for Climate Emulator testing

Please note that the data described below is added to the data folder as source data as well as parts of it are split across different subfolders concentration, emission, forcing, tempearture. Thus we do not describe each folder spearately, but just the source data.

For the calibration of the CE we use the benchmark data that was added to this folder.

The data are based on montly mean global mean surface temperature data from the CMIP5 archive, as downloaded on March 1, 2021, from http://iacweb.ethz.ch/staff/beyerleu/cmip5/. The site offers an easy to use mirror to the full CMIP5 archive at https://esgf-node.llnl.gov/search/cmip5/.

The monthly mean data have been aggregated to annual means using cdo (climate data operators, https://code.mpimet.mpg.de/projects/cdo/). The data comes in netcdf format.

CMIP5 input data, historical and RCPs

We used carbon emissions (GtC, fossile, other, total) and concentrations (CO2) as used in CMIP5 historical and RCP scenarios (see Taylor et al. (2012)).

The data from Taylor et al. (2012) can be downloaded from the Potsdam Institute for Climate Impact Reasearch from following URL. Note that RCP3PD at PIK corresponds to RCP26. Likewise RCP6 at PIK corresponds to RCP60.

The following files contain annual carbon emissions from 1750 to 2500 for the different RCPs:

- RCP26_EMISSIONS.csv
- RCP45_EMISSIONS.csv
- RCP60_EMISSIONS.csv
- RCP85_EMISSIONS.csv

The following files contain annual mean CO2 concentrations from 1750 to 2500 for the different RCPs:

- RCP26_MIDYR_CONC.DAT
- RCP45_MIDYR_CONC.DAT
- RCP60_MIDYR_CONC.DAT
- RCP85_MIDYR_CONC.DAT

For simplicity, we added these datasets to this folder.

Additionally we used the land emissions data from Gasser et al. (2020).

The follwoing file contains the data on land emissions: - Gasser_et_al_2020_best_guess.nc

The data can be downloaded from the International Institute for Applied Systems Analysis from following URL. We added these dataset to this folder.

Pulse

We used pulse data from Joos et al. (2013).

The following files contain pre-processed timeseries of all the models under PD: - IRF_PD100_SMOOTHED_AFCUM.dat - IRF_PD100_SMOOTHED_CO2.dat - IRF_PD100_SMOOTHED_FABCUM.dat - IRF_PD100_SMOOTHED_FASCUM.dat - IRF_PD100_SMOOTHED_OHC.dat - IRF_PD100_SMOOTHED_SSLR.dat - IRF_PD100_SMOOTHED_T.dat

The following files contain pre-processed timeseries of all the models under PI: - IRF_PI100_SMOOTHED_AFCUM.dat - IRF_PI100_SMOOTHED_CO2.dat - IRF_PI100_SMOOTHED_FABCUM.dat - IRF_PI100_SMOOTHED_FASCUM.dat - IRF_PI100_SMOOTHED_OHC.dat - IRF_PI100_SMOOTHED_SSLR.dat - IRF_PI100_SMOOTHED_T.dat

The data can be downloaded from the University Bern from following URL.

For simplicity, we added these datasets to this folder.

Zero Emission Commitment

We used ZEC data from Jones et al. (2019) and Macdougall et al. (2020).

The description can be found at the University of Victoria page URL and the data can be downloaded from the following URL.

The list of the files with the data can be found here.

For simplicity, we added these datasets to this folder.

Pattern scaling data

For pattern scaling we used three datasets.

First, we used the **IPCC AR6 WGI v4 reference regions** from Iturbide et al. (2020). The data can be found in this repository.

Second, we used a **library of temperature and precipitation patterns** from this repository from Lynch et al. (2017), Kravitz et al. (2016).

Third, we used **ERA5 data** for the global climate and weather. The data can be found at the page of Climate Data Store of Copernicus Climate Change Service URL.

For simplicity, we added these datasets to the follwoing folders: - folder - tas_climmean_era5_1991-2020_g025.nc - tas_climmean_HadGEM2-ES_minus_era5_1991-2020_g025.nc - tas_climmean_MPI-ESM-LR_minus_era5_1991-2020_g025.nc

- folder
    - IPCC-WGI-reference-regions-v4_coordinates.csv
- folder
    - PATTERN_tas_ANN_HadGEM2-ES_rcp85.nc
    - PATTERN_tas_ANN_MPI-ESM-LR_rcp85.nc
- folder
    - ipcc_regions.csv
    - PATTERN_tas_ANN_HadGEM2-ES_rcp85.nc
    - PATTERN_tas_ANN_MPI-ESM-LR_rcp85.nc
    - tas_climmean_era5_1991-2020_g025.nc
- folder (IPCC AR6 WGI v4 reference regions data). The list of the files with the data can be found here.

Computational requirements

Software requirements

The basic dependencies are tensorflow==2.3, hydra-core >= 1.1 and tensorboard (for monitoring). For a full set of dependencies see the environment.yaml - this includes the development environment as well (Spyder 3).

- We provide implementations that use python 3.10.
- For the The basic dependencies are Tensorflow==2.x, hydra and Tensorboard (for monitoring).

- The file `requirements.txt` lists the detailed dependencies. Please run “pip install -r requirements.txt” as the first step. See here for further instructions on creating and using the `requirements.txt` file.

Controlled randomness

The random seed for our computations in *Section 4 - Numerical Experiments* is set at `Replication_Building_Emulators/DEQN/config/config.yaml`, line 10.

Memory and runtime requirements

- To solve one IAM model as discussed in *Section 4 - Numerical Experiments* until full convergence, it requires about 15 min on an ordinary laptop. All those models presented in the paper were solved using our DEQN library, which we ran on an 8-core Intel compute node on https://nuvolos.cloud with 64GB of RAM, and 100Gb of fast local storage (SSD).
- All the postprocessing codes (to produce the summary statistics, plots, and so forth) were run on an ordinary 4-core Intel-based laptop with Ubuntu version 18.04.5 LTS and consume typically few seconds to run.
- The approximate time needed to reproduce all the analyses for this paper on a standard (year 2025) desktop machine is 1-3 days of human time.

Replication

- This section provides instructions on how to replicate the numerical results of the article. Note that in this readme, we only provide the basic steps to obtain the main results (e.g., for the multi-model mean of the CDICE model). Highly granular instructions on how to compute all the corner cases of the article are provided in readmes in the respective sub-folders.
- The optimal optimal order of running the computer code to replicate the results in article are as follows.
  1. Run the instructions listed in the subsection `1. Replication of Section 3: A Calibrated Climate Emulator`. This set of instructions will allow you to replicate the results of section 3 of the article.
  2. Run the instructions listed in `2. Replication of Section 4: Numerical Experiments`. This set of instructions will allow you to replicate the results of section 4 of the article.
  3. Run the instructions listed in `3. Replication of Section 5: Linking Global to Local Temperatures and Damages`. This set of instructions will allow you to replicate the results of section 5 of the article.

1. Replication of Section 3: A Calibrated Climate Emulator

In this section, we provide the basic instructions on how to compute the results presented in section 3 of the article.

First, go to the following folder:

```
$ cd <PATH to the repository>Replication_Building_Emulators/figures_replication/figures_3-4_14-25
```

Script `solver.py` fits the 3SR and 4PR carbon cycle models to benchmark datasets, performing a grid search over penalty parameters. To run the fitting procedure please perfrom the following: `bash python3 solver.py <conditions: PI|PD> <T:int> <benchmark_sel>` * **PI**: Preindustrial benchmark set * **PD**: Present-day benchmark set * <T>: Time horizon in years * `<benchmark_sel>`: One benchmark model name from the allowed set for the selected condition

For example,

```
python3 solver.py PD 250 CLIMBER2
```

Results form every model are stored in this folder.

For visualizations: - `python3 fig_benchmark_pulse.py` Plot benchmark pulse decays for diffrenent models; - `python3 fig_RCP_RF.py` Plots related to the RCP scenerios; - `python3 fig_analysis.py` Analysie emulator hyperpmanters and solve extreama paramters for the MMM fitted emulator; - `python3 fig_sim.py` Different simulations using using the extrama parmater of MMM fitted emulator.

- More details are provided in this general readme and this readme that is specific to the section 3.

2. Replication of Section 4: Numerical Experiments

In this section, we provide the basic instructions on how to compute the results presented in section 4 of the article.

First, go to the following folder:

```
$ cd <PATH to the repository>Replication_Building_Emulators/figures_replication/figures_5-8_26-31
```

First, make sure you are at the root directory of DEQN by changing path to the following sub-directory:

```
$ cd <PATH to the repository>Replication_Building_Emulators/DEQN
```

To start the computation from scratch, change the specifications in the config file (config/config.yaml) to the particular model of interest while leaving the other entries untouched. The following entries will have to be adjusted:

```
defaults:
  - constants: cdice
  - net: cdice
  - optimizer: cdice
  - run:  cdice
  - variables:  cdice


MODEL_NAME:   XXX
```

XXX stands for the specific name of the model, the list of the models is presented here .

Thereafter, execute:

```
python run_deepnet.py
```

To analyze the the raw results for the particular model solved, you need to perform two steps:

```
$ export USE_CONFIG_FROM_RUN_DIR=$ cd export USE_CONFIG_FROM_RUN_DIR=<PATH_TO_THE_FOLDER>/Replication_Bu
```

```
$ python post_process_XXX.py STARTING_POINT=LATEST hydra.run.dir=$USE_CONFIG_FROM_RUN_DIR
```

where XXX stands for the postprocessing routine specific to the model.

For examplem, the setting for solving the 3SR model calibrated under PI for optimal mitigation **cdice_3sr_pi_opt** model is the following:

```
  - constants: cdice
  - net: cdice
  - optimizer: cdice
  - run:  cdice
  - variables: cdice

MODEL_NAME:  cdice_3sr_pi_opt
```

To analyze this model, you need to run the following:

```
 export USE_CONFIG_FROM_RUN_DIR=<PATH_TO_THE_FOLDER>/Replication_Building_Emulators/DEQN/runs/cdice_3sr_
```

```
python post_process_3sr.py STARTING_POINT=LATEST hydra.run.dir=$USE_CONFIG_FROM_RUN_DIR
```

To replicate the remaining results for individual models discussed in *Section 4 - Numerical Experiments*, the instructions given here.

- More details on the replication of the figures are provided in this readme.

3. Replication of Section 5: Linking Global to Local Temperatures and Damages

In this section, we provide the basic instructions on how to compute the results presented in section 3 of the article.

First, go to the following folder:

```
$ cd <PATH to the repository>Replication_Building_Emulators/figures_replication
```

Folders `figure_9`, `figure_10`, `figure_11`, `figure_12`, `figure_13` contain all the scripts and data for replication of respective figures.

- More details are provided in this readme.

Authors

- Aryan Eftekhari (Universita della Svizzera italiana, Faculty of Informatics, and the University of Lausanne, Department of Economics)
- Doris Folini (ETH Zuerich, Institute for Atmospheric and Climate Science)
- Felix Kuebler (the University of Zuerich, Department for Banking and Finance, and Swiss Finance Institute)

- Aleksandra Friedl (the University of Lausanne, Department of Economics, and ifo Institute)
- Simon Scheidegger (the University of Lausanne, Department of Economics)
- Olaf Schenk (Universita della Svizzera italiana, Faculty of Informatics)

Citation

Please cite Building Interpretable Climate Emulators for Economics in your publications if it helps your research:

```
@article{eftekhari2024building,
  title={Building interpretable climate emulators for economics},
  author={Eftekhari, Aryan and Folini, Doris and Friedl, Aleksandra and K{\"u}bler, Felix and Scheidegge
  journal={arXiv preprint arXiv:2411.10768},
  year={2024}
}
```

Support

This work was generously supported by grant from the Swiss National Science Foundation under project ID “Can economic policy mitigate climate change?”.