



ANALYSE AUTOMATISÉE DE L'INFORMATION CLIMATIQUE OUTIL D'ANALYSE PYTHON

Yannick Rousseau
Spécialistes en scénarios
et services climatiques

document: 14 septembre 2020
code: scen_workflow_afr (v1.0.3)

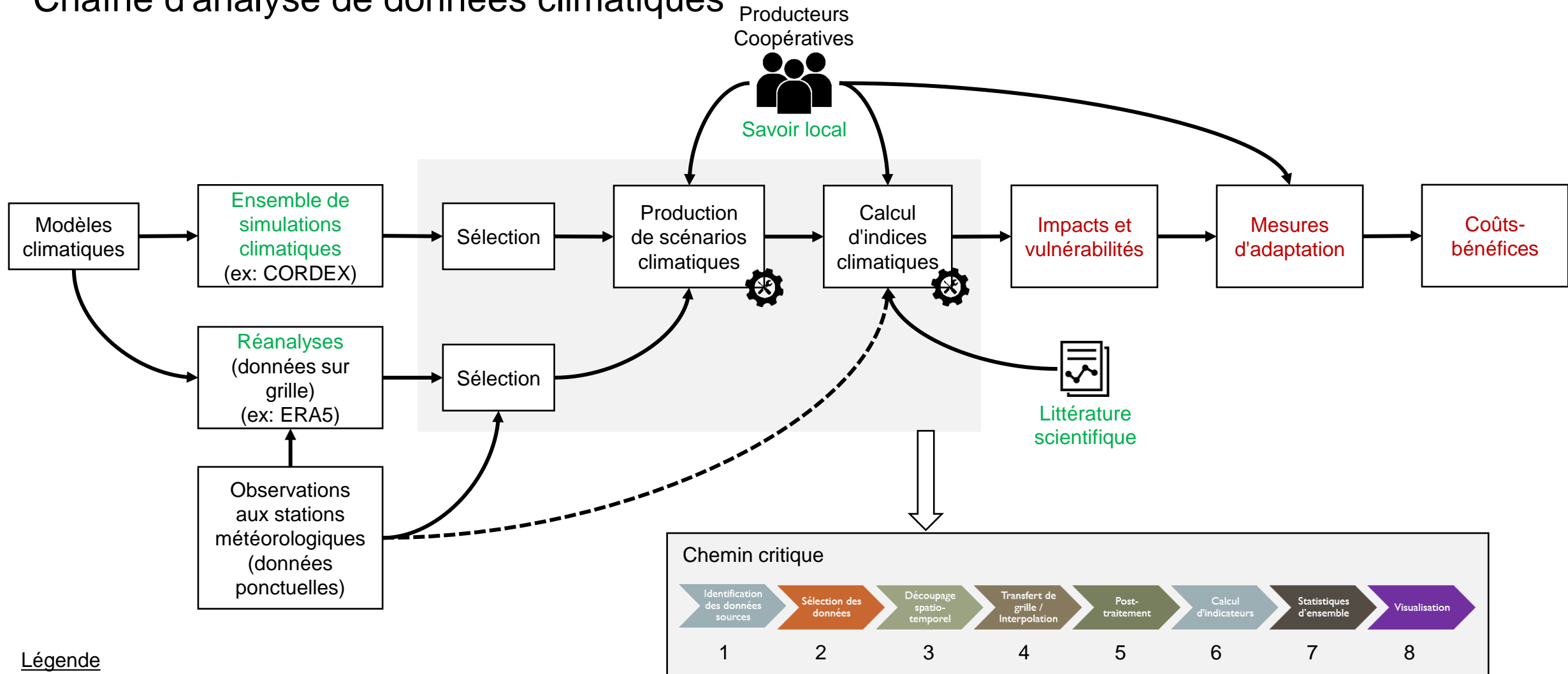
1. Présentation des participants (15 min.)
2. Processus et concepts en scénarisation climatique (15 min.)
 - Revue du 'chemin critique'
 - Revue des concepts essentiels
3. Code Python (30 min.)
 - Utilité et emplacement
 - Fichiers et répertoires
 - Configuration (stations météorologiques; variables et indices climatiques)
4. Environnement de développement (60 min.)
 - Installation d'applications, du code source et des librairies nécessaires
 - Inspection du code et diagnostic d'une anomalie
5. Exécution du script (15 min.)
 - Procédure
 - Éléments visuels
6. Prochaines étapes (15 min.)

ANALYSE DE L'INFORMATION CLIMATIQUE

2. PROCESSUS ET CONCEPTS



Chaîne d'analyse de données climatiques



Légende
Intrant
Extrant

Chemin critique (8 étapes)



Étapes effectuées en 2019 (1 à 5)

- Identification des besoins (ex: scénarios d'émissions, variables climatiques)
- Revue des concepts en scénarisation climatique
- Accès aux données observées à des stations météorologiques (facultatif)
- Code Python pour la production des scénarios climatiques

Étapes effectuées en 2020 (6 et 8, en partie)

- Code Python pour le calcul d'un indice climatique
- Code Python pour la production automatisée de diagrammes et de cartes

Étapes spécifiques à tout projet

- Ajout et calcul d'indices climatiques (nécessite les seuils)
- Génération d'éléments visuels supplémentaires et interprétation des résultats
- Utilisation de la scénarisation climatique pour accomplir les objectifs d'un projet

Scénario climatique, selon Ouranos:

Représentation simplifiée et plausible du climat futur, construite à partir de simulations climatiques.

Les projections climatiques servent de matériaux bruts.

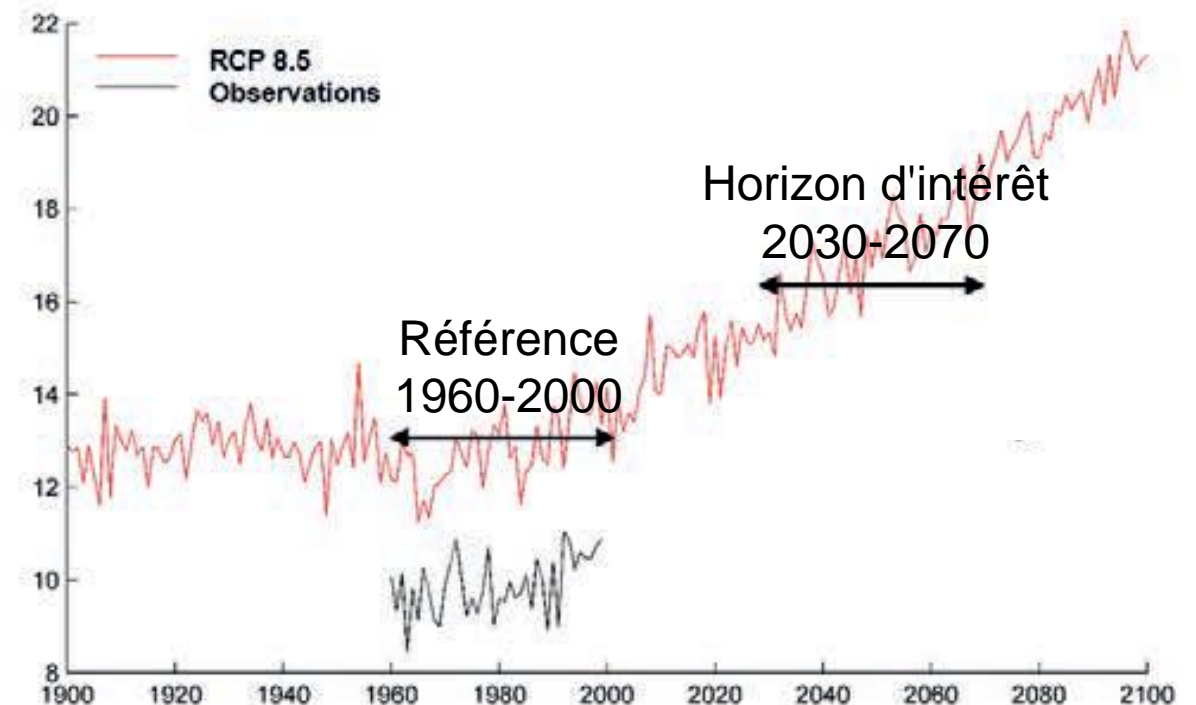
Ils exigent souvent de l'information additionnelle, comme le climat actuel observé.

(adapté du guide des scénarios climatique d'Ouranos de 2016)

Selon le GIEC:

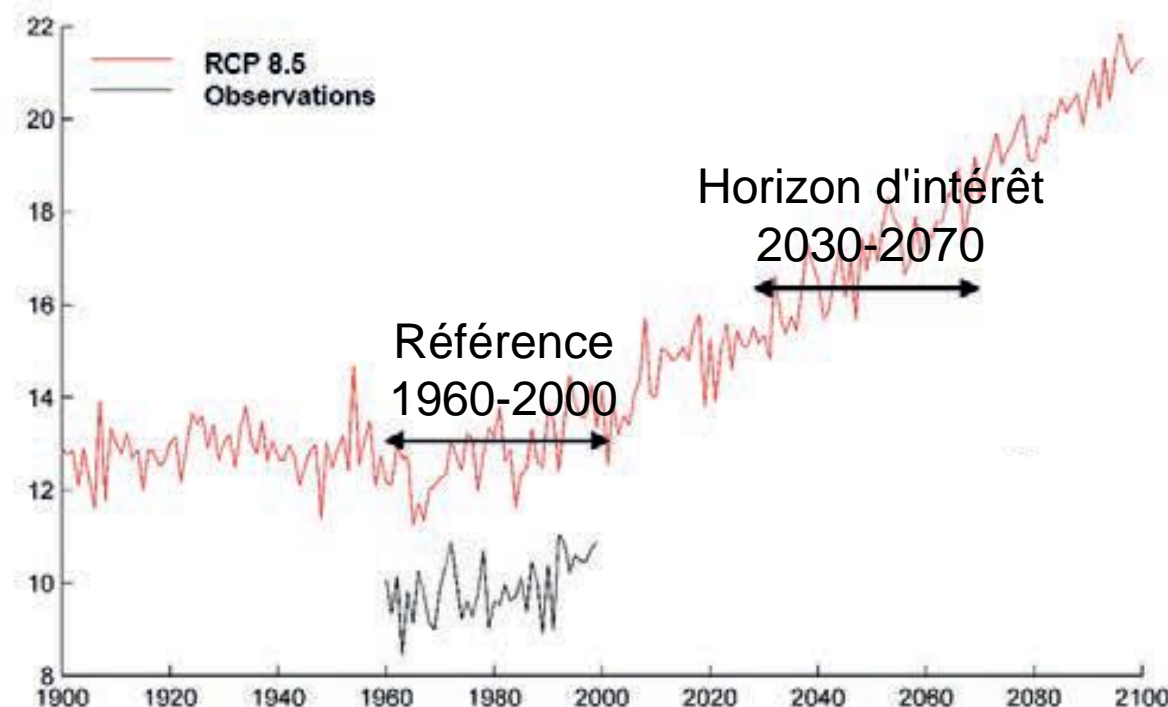
Plausible representation of future climate that has been constructed for explicit use in investigating the potential impacts of anthropogenic climate change.

Make use of climate projections, by manipulating model outputs and combining them with observed climate data (IPCC, 2001, chapitre13)



Points importants

- Emplacement précis
Ex: Ville ou région
- Inclut le passé et le futur.
- Ajustement de la sortie des modèles climatiques
nécessaire pour correspondre aux observations:
Correspondance entre séries temporelles et limites
connues des espèces et matériaux.
Ajustement/correction du biais.
Mise à l'échelle statistique.
- Variable spécifique et concrète
Ex: Température maximale journalière
- Scénario d'émissions de GES spécifique
Ex: RCP 8.5

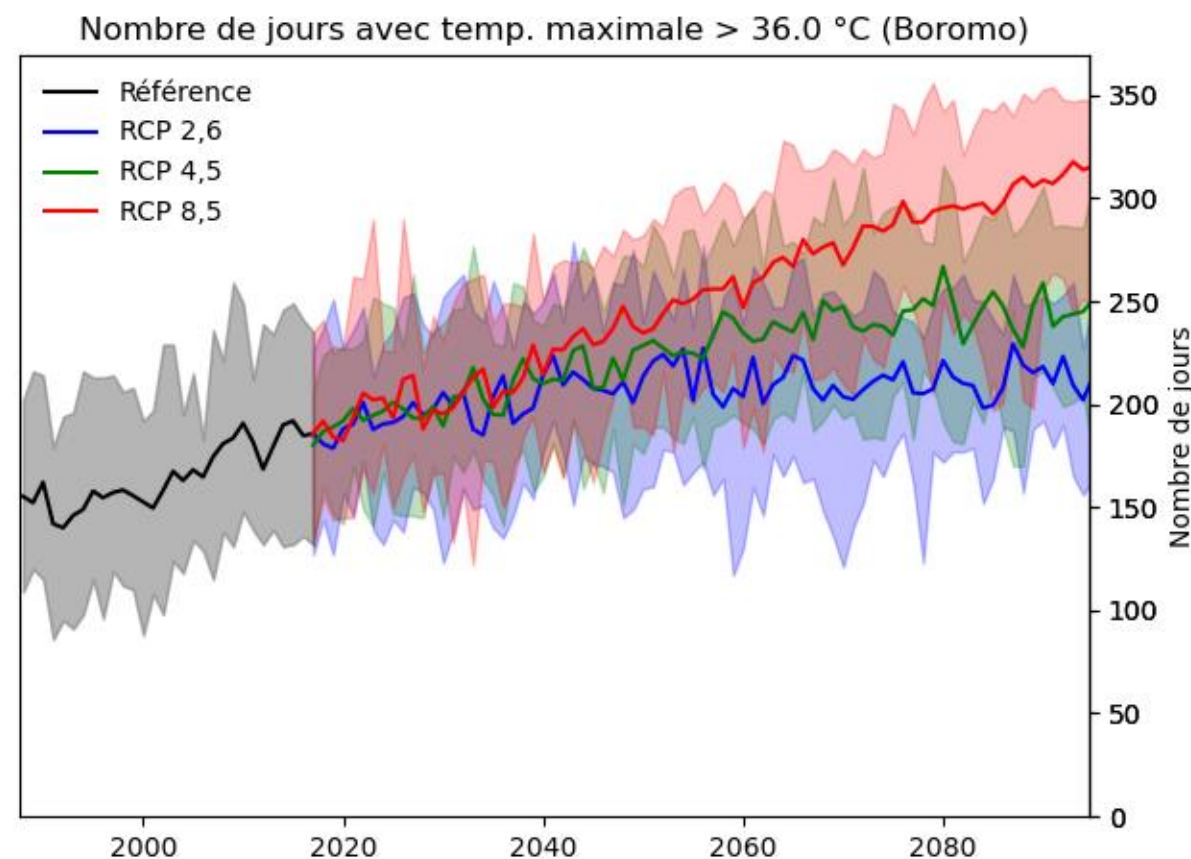


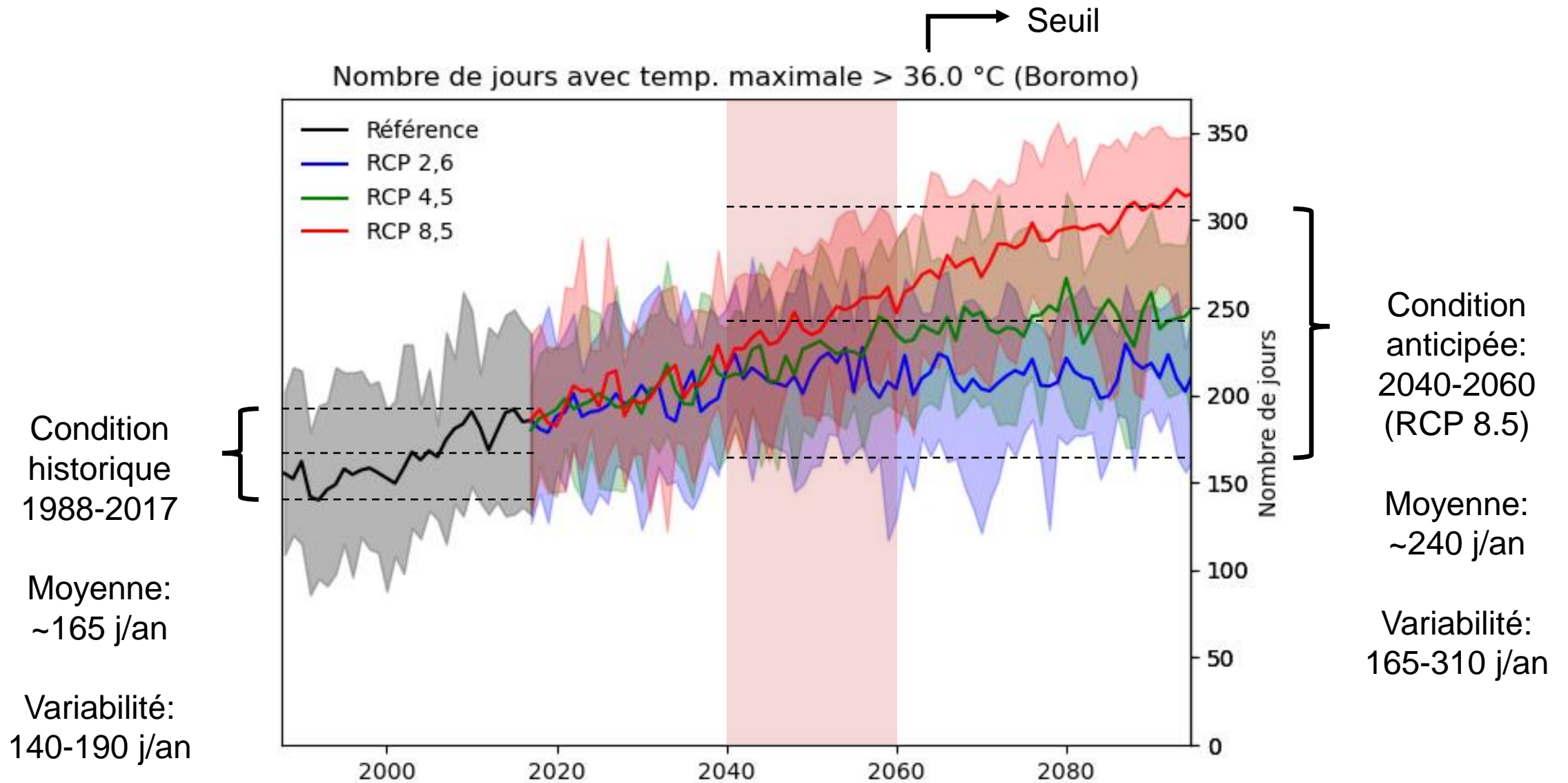
Indice climatique

- Propriété du climat qui n'est pas mesurée sur le terrain ou calculée par les modèles climatiques, mais qui est plutôt calculée ou dérivée de variables climatiques plus simples (ex: température, précipitations) (donneesclimatiques.ca).
- Ex: degrés-jours de croissance, cycles de gel-dégel et l'indice de sécheresse.

Points importants

- Associé à la notion de rendement
Ex: conditions idéales, adéquates, nuisibles
- Utilisé pour la prise de décision, même sans tenir compte du changement climatique.
- Permet d'imaginer les impacts du changement climatique de façon concrète et ciblée (plus pratique que la concentration en CO₂ ou la valeur d'une variable).
- Peut combiner plusieurs variables.
Ex: température et précipitations.





Seuil associé à un indice climatique

- Valeur d'une variable climatique à partir de laquelle des effets se font sentir sur une activité socio-économique ou sur la santé humaine.
- Les effets peuvent être positifs ou négatifs.
- Ex: une période de canicule à Montréal survient lorsque la température maximale journalière est supérieure à 33°C et la température minimale est supérieure à 22°C pendant trois jours

(Institut national de la Santé publique du Québec).

Points importants

- Plusieurs sources d'information
Ex: littérature scientifique, guides normatifs, savoir local, données météorologiques.
- Associé à un bénéfice ou à une conséquence précise.
- Potentiellement plusieurs seuils pour un seul indice climatique.
- Sa valeur peut être difficile à déterminer.
- Certaine dose d'incertitude.
- Information précieuse/essentielle.

ANALYSE DE L'INFORMATION CLIMATIQUE

3. CODE PYTHON



Utilité

- Traitement automatisé d'un large volume de données climatiques (2-5 To pour un seul projet)
Ex1: Sortie des modèles climatiques
Ex2: Observations météorologiques
Ex3: Réanalyses (optionnel)
- Ajustement de la sorties de modèles climatiques à l'aide d'observations météorologiques
- Considération d'un ensemble de simulations climatiques
- Utilisation d'un code Python développé à l'interne (Ouranos) et externe
- Haute performance (bibliothèques xclim, xarray, dask, xesmf, numpy, pandas, matplotlib)

Hébergement

- Projet privé sur Github (pour l'instant)
- github.com/Ouranosinc/scen_workflow_afr

Environnement de développement

- PyCharm recommandé (<https://www.jetbrains.com/pycharm/>)
- Linux Ubuntu

git - gestion de code source (privé)

- Accès par le site web
https://github.com/Ouranosinc/scen_workflow_afr
- Création d'un compte
Code d'usager transmis à l'administrateur du compte Github d'Ouranos
- Mise à jour à partir du terminal

Terminal

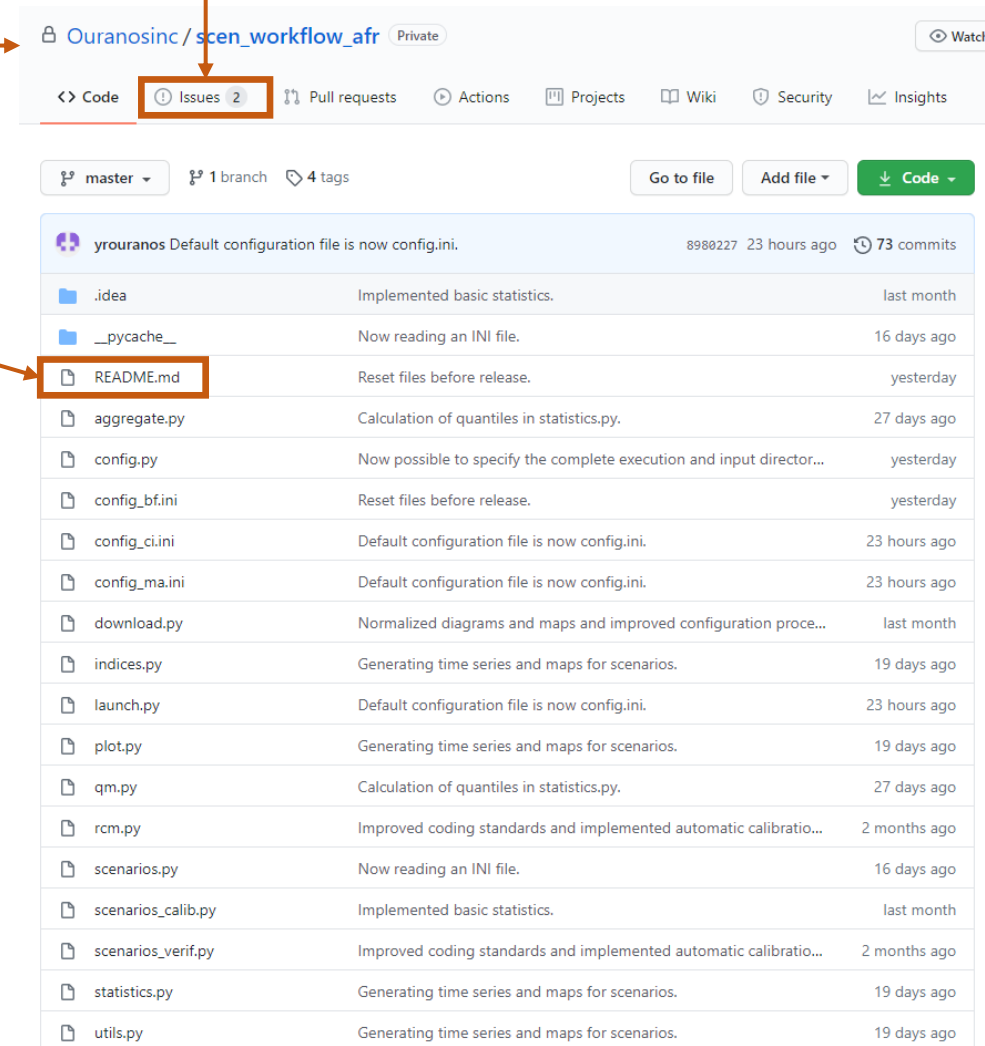
```
# Installer git.
$ sudo apt-get install git

# Télécharger le code source.
$ cd ~/Documents/dev/
$ git clone https://github.com/<code_usager>/scen_workflow_afr.git

# Obtenir les modifications les plus récentes.
$ cd scen_workflow_afr
$ git pull

# Soumettre une amélioration/modification.
$ cd scen_workflow_afr
$ git commit -a -m "Ajout de l'indice climatique X."
$ git push
```

Anomalies et demandes
de changements



Ouranosinc / scen_workflow_afr (Private)

<> Code Issues 2 Pull requests Actions Projects Wiki Security Insights

master 1 branch 4 tags Go to file Add file Code

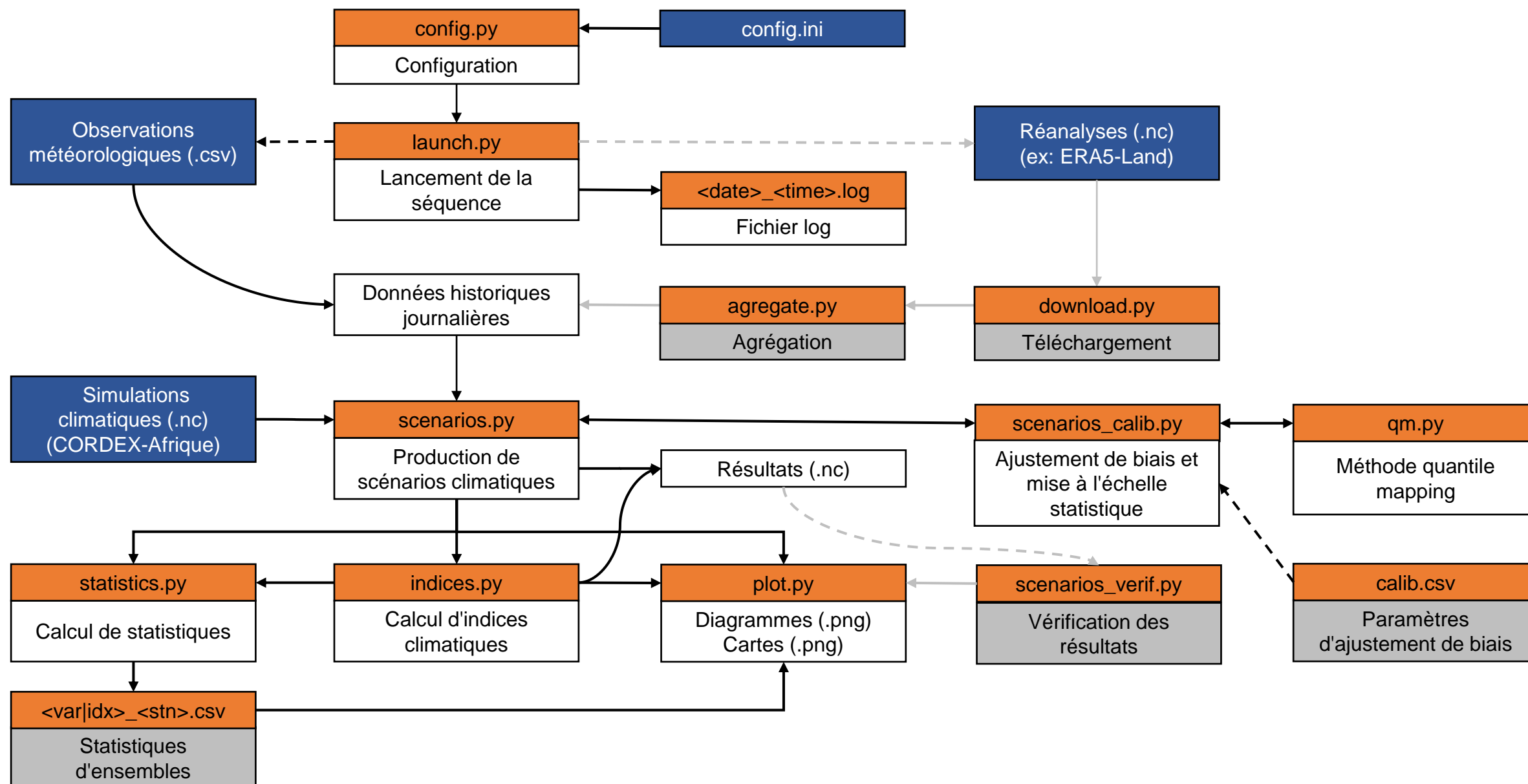
yrouanos Default configuration file is now config.ini. 8980227 23 hours ago 73 commits

.idea	Implemented basic statistics.	last month
__pycache__	Now reading an INI file.	16 days ago
README.md	Reset files before release.	yesterday
aggregate.py	Calculation of quantiles in statistics.py.	27 days ago
config.py	Now possible to specify the complete execution and input director...	yesterday
config_bf.ini	Reset files before release.	yesterday
config_ci.ini	Default configuration file is now config.ini.	23 hours ago
config_ma.ini	Default configuration file is now config.ini.	23 hours ago
download.py	Normalized diagrams and maps and improved configuration proce...	last month
indices.py	Generating time series and maps for scenarios.	19 days ago
launch.py	Default configuration file is now config.ini.	23 hours ago
plot.py	Generating time series and maps for scenarios.	19 days ago
qm.py	Calculation of quantiles in statistics.py.	27 days ago
rcm.py	Improved coding standards and implemented automatic calibratio...	2 months ago
scenarios.py	Now reading an INI file.	16 days ago
scenarios_calib.py	Implemented basic statistics.	last month
scenarios_verif.py	Improved coding standards and implemented automatic calibratio...	2 months ago
statistics.py	Generating time series and maps for scenarios.	19 days ago
utils.py	Generating time series and maps for scenarios.	19 days ago

Fichiers

- | | |
|----------------------|---|
| • config.py | Paramètres globaux |
| • launch.py | Séquence de traitement |
| • download.py | Téléchargement (réanalyses, optionnel) |
| • aggregate.py | Agrégation de données (réanalyses, optionnel) |
| • scenarios.py | Production de scénarios climatiques (ex: ajustement de biais, mise à l'échelle) |
| • scenarios_calib.py | Recherche automatisée des paramètres optimaux |
| • scenarios_verif.py | Vérification des résultats (optionnel) |
| • qm.py | Implémentation de la méthode de quantile mapping |
| • indices.py | Calcul d'indices climatiques |
| • statistics.py | Calcul des statistiques d'ensemble |
| • plot.py | Génération d'éléments visuels (diagrammes et cartes) |
| • rcm.py, utils.py | Diverses fonctions utilitaires n'entrant dans aucune catégorie spécifique |

Les lignes grisées correspondent à du code Python qui n'est pas utilisé dans le présent projet.



config.py et config.ini

- Constantes et variables
- Structure des répertoires
- Options d'analyse

launch.py

- Variables spécifiques au projet
- Structure des répertoires
- Exceptions (simulations et variables pas encore compatibles avec le script)
- Lancement de la séquence d'analyse

launch.py

```
load_params(p_ini)                # Chargement des paramètres.

download.run()                    # Téléchargement.
aggregate.run()                   # Agrégation (si nécessaire).

scen_calib.init_calib_params()    # Initialisation des paramètres pour l'ajustement de biais.
scenarios.run()                   # Production de scénarios climatiques (pré et post-traitement,
                                # interpolation).

scen_calib.bias_correction(<stn>, <var>, <sim>) # Ajustement de biais (modes: désactivé, fichier de paramètres, manuel ou
                                # automatisé; méthode statistique).

indices.run()                     # Calcul d'indices climatiques (indices et seuils).
statistics.run()                  # Calcul de statistiques d'ensembles (scénarios et indices).
```

ANALYSE DE L'INFORMATION CLIMATIQUE

3. CODE PYTHON > CONFIGURATION (VARIABLES)

clé

config.ini (utilisation d'observations)

[PROJECT] ← **section**
country:"burkina"
project:"pcci" ← **valeur**

[OBSERVATIONS]
obs_src:"anam"
file_sep:";"
stns:["bereba","boromo","boura","diebougou","farakoba","gao","gaoua","hounde","kassoum","koumbia","leo","nasso","po","sapouy","valleedukou"]

[CONTEXT]
rcps:["rcp26","rcp45","rcp85"]
per_ref=[1988,2017]
per_fut=[1988,2095]
per_hors=[[2021,2040],[2041,2060],[2061,2080]]
variables_cordex:["tas","tasmin","tasmax","pr","uas","vas"]

[SCENARIOS]
opt_scen=True
opt_scen_regrid=False
lon_bnds=[-6,3]
lat_bnds=[8,16]
sim_excepts:["RCA4_AFR-44_ICHEC-EC-EARTH_rcp85",
"RCA4_AFR-44_MPI-M-MPI-ESM-LR_rcp85",
"HIRHAM5_AFR-44_ICHEC-EC-EARTH_rcp45.nc",
"HIRHAM5_AFR-44_ICHEC-EC-EARTH_rcp85.nc"]
var_sim_excepts:["pr_RCA4_AFR-44_CSIRO-QCCCE-CSIRO-Mk3-6-0_rcp85.nc",
"tasmin_REMO2009_AFR-44_MIROC-MIROC5_rcp26.nc"]

[CALIBRATION]
opt_calib=True
opt_calib_auto=False
opt_calib_bias=True
opt_calib_bias_meth:"rrmse"

[INDICES]
opt_idx=True
idx_names:["tx_days_above"]
idx_threshs=[[36.0]]

[STATISTICS]
opt_stat=True
stat_quantiles=[1.00, 0.99, 0.75, 0.50, 0.25, 0.01, 0.00]

[VISUALIZATION]
opt_plot=True
Source: <https://datacatalog.worldbank.org/dataset/burkina-faso-administrative-boundaries-2017>
d_bounds:"bf_boundaries.geojson"

[ENVIRONMENT]
n_proc=4
d_exec:"/exec/yrousseau/"
d_proj:"/media/yrousseau/ROCKET-XTRM/scenario/external_data/CORDEX-AFR/"

config.ini (utilisation de réanalyses)

```
[PROJECT]
country:"coteivoire"
project:"adaptcoop"

[OBSERVATIONS]
obs_src:"era5_land"

[CONTEXT]
rcps:["rcp45","rcp85"]
per_ref=[1981,2010]
per_fut=[1981,2095]
per_hors=[[2021,2050],[2041,2070]]
variables_cordex:["tas","tasmin","tasmax","pr","uas","vas",
"evapsbl","evapsblpt","rsds","ps","huss"]
variables_ra:["t2m","tp","u10","v10","e","pev","ssrd","sp","d2m"]

[DATA]
opt_download=True
lon_bnds_download=[-30,30]
lat_bnds_download=[-15,40]
obs_src_username=""
obs_src_password=""
opt_aggregate=True

[SCENARIOS]
opt_scen=True
opt_scen_regrid=False
lon_bnds=[-9,-2]
lat_bnds=[4,11]
sim_excepts:["RCA4_AFR-44_ICHEC-EC-EARTH_rcp85",
"RCA4_AFR-44_MPI-M-MPI-ESM-LR_rcp85",
"HIRHAM5_AFR-44_ICHEC-EC-EARTH_rcp45.nc",
"HIRHAM5_AFR-44_ICHEC-EC-EARTH_rcp85.nc"]
```

```
var_sim_excepts:[
"pr_RCA4_AFR-44_CSIRO-QCCCE-CSIRO-Mk3-6-0_rcp85.nc",
"tasmin_REMO2009_AFR-44_MIROC-MIROC5_rcp26.nc"]

[CALIBRATION]
opt_calib=True
opt_calib_auto=False
opt_calib_bias=True
opt_calib_bias_meth:"rrmse"

[INDICES]
opt_idx=True
idx_names:["tx_days_above"]
idx_threshs=[[36.0]]

[STATISTICS]
opt_stat=True
stat_quantiles=[1.00, 0.99, 0.75, 0.50, 0.25, 0.01, 0.00]

[VISUALIZATION]
opt_plot=True
# Source: https://datacatalog.worldbank.org/dataset/cote-divoire-
administrative-boundaries-2016
d_bounds:"ci_boundaries.geojson"

[ENVIRONMENT]
n_proc=4
d_exec:"/exec/yrousseau/"
d_proj:"/media/yrousseau/ROCKET-XTRM/scenario/external_data/
CORDEX-AFR/"
d_ra_raw:"/media/yrousseau/ROCKET-XTRM
/scenario/external_data/ecmwf/era5_landhour/"
d_ra_day:"/media/yrousseau/wd/scenario/external_data/ecmwf/era5_land/day
/"
```


config.py

```
# Variables climatiques.
variables_cordex      = ["?"]      # Variables CORDEX utilisées dans l'analyse.
var_cordex_*          = "?"        # Nom des variables CORDEX.
variables_ra          = ["?"]      # Variables des données de réanalyse utilisées dans l'analyse.
var_era_*             = "?"        # Nom des variables ERA5 et ERA5-Land.

# Scénarios, stations, périodes.
rcps                  = ["?"]      # Scénarios d'émissions utilisés.
stns                   = ["?"]      # Stations d'observation des données météorologiques.
per_ref, per_fut      = [?]        # Périodes de référence et future.
per_hors               = [[?:?]]   # Horizons climatiques.

# Correction de biais.
nq_default            = ?          # Valeur par défaut: Nombre de quantiles.
up_qmf_default        = ?          # Valeur par défaut: Limite supérieure de la fonction de quantile mapping.
time_win_default      = ?          # Valeur par défaut: Nombre de jours avant et après une date donnée (semi-largeur de la fenêtre).
nq_calib              = [?]        # Liste de valeurs de 'nq_default' à tester.
up_qmf_calib          = [?]        # Liste de valeurs de 'up_qmf_default' à tester.
time_win_calib        = [?]        # Liste de valeurs de 'time_win_default' à tester.
```

config.py

```

# Scénarios climatiques.
opt_scen                = True                # Si True, produire les scenarios climatiques.
opt_scen_read_obs_netcdf = True                # Si True, convertir les observations en fichiers NetCDF.
opt_scen_extract        = True                # Si True, forcer une extraction spatiotemporelle.
opt_scen_itp_time       = True                # Si True, interpoler temporellement lors de l'extraction.
opt_scen_itp_space      = True                # Si True, interpoler spatialement lors de l'extraction.
opt_scen_regrid         = False               # Si True, utiliser refaire le maillage lors de l'interpolation.
                                     # Sinon, prendre la valeur associée à la cellule la plus près.

opt_scen_preprocess     = True                # Si True, forcer le pré-traitement.
opt_scen_postprocess    = True                # Si True, forcer le post-traitement.

# Ajustement de biais.
opt_calib               = True                # Si True, ajuster le biais pour quelques valeurs des paramètres nq, up_qmf et time_win.
opt_calib_auto          = True                # Si True, trouver les valeurs optimales des paramètres nq, up_qmf et time_win.
opt_calib_bias          = True                # Si True, quantifier l'erreur de l'ajustement du biais.
opt_calib_bias_meth     = "rrmse"             # Méthodes de quantification de l'erreur.
opt_calib_bias_meth_r2  = "r2"                # Coefficient of détermination.
opt_calib_bias_meth_mae = "mae"               # Mean absolute error.
opt_calib_bias_meth_rmse = "rmse"             # Root mean square error.
opt_calib_bias_meth_rrmse = "rrmse"           # Relative root mean square error.
opt_calib_qqmap         = True                # Si true, calculer qqmap.

# Indices climatiques.
opt_idx                 = True                # Si True, calculer les indices climatiques.

# Statistiques d'ensemble.
opt_stat                = True                # Si True, calculer les statistiques d'ensemble.
stat_quantiles          = [1.00, 0.99, 0.75, 0.50, 0.25, 0.01, 0.00]

# Diagrammes et cartes.
opt_plot                = True                # Si True, produire les diagrammes et cartes.

```

Tâches

- Configuration du projet (contexte et système).
- Ajout de variables (ex: CORDEX) et d'indices climatiques.
- Emplacements clés identifiés dans le code.

```
*.py
# =====
# TODO.CUSTOMIZATION.BEGIN
# Instructions
# =====

...

# =====
# TODO.CUSTOMIZATION.END
# =====
```

Documentation

- <https://xclim.readthedocs.io/en/stable/indicators.html>
59 indices atmosphériques
- Indices plus complexes nécessitent du développement.

Indice 'prototype'

config.py & launch.py

```
# Activation du calcul des indices climatiques et déclaration des indices et seuils.
cfg.opt_idx      = True
cfg.tx_days_above = "tx_days_above"          # Nom de l'indice climatique.
cfg.idx_names    = [cfg.tx_days_above]        # Indices climatiques utilisés.
cfg.idx_threshs  = [[36]]                    # Seuils associés aux indices climatiques.
```

indices.py

```
# Sélection des variables (CORDEX) nécessaires au calcul de l'indice climatique.
```

```
vars = []
if idx_name == cfg.idx_tx_days_above:
    vars = [cfg.var_cordex_tasmax]
```

```
# Calcul d'un indice: nombre de jours par an où Tmax > seuil.
```

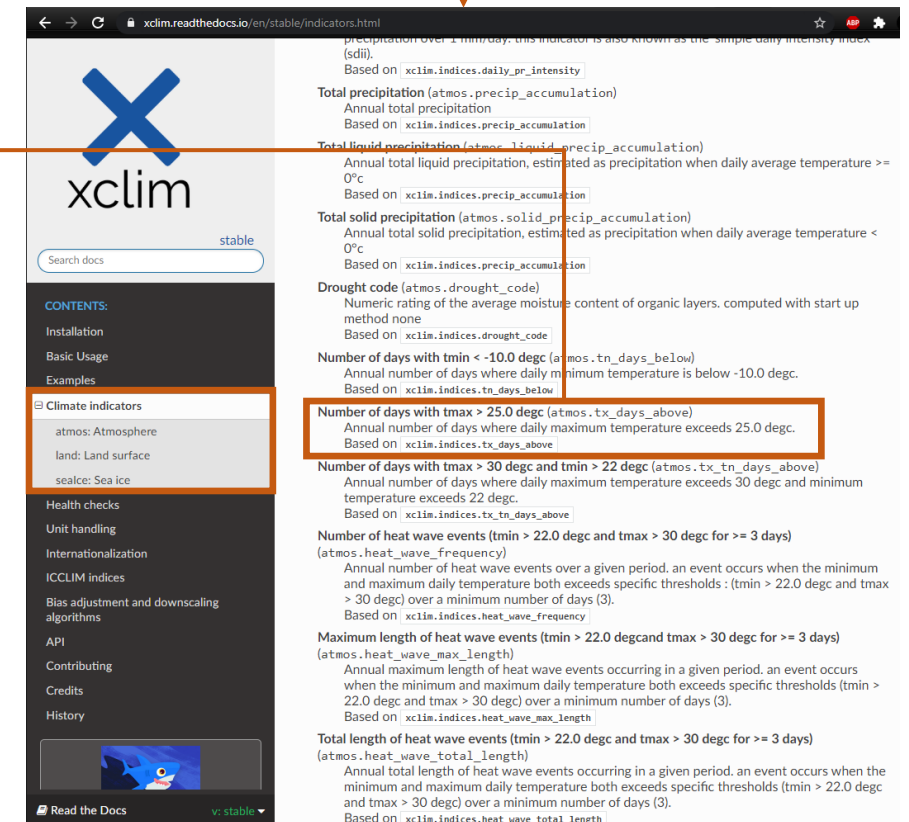
```
ds_idx = None
```

```
if idx_name == cfg.idx_tx_days_above:
    ds_scen_tasmax = ds_scen[0][cfg.var_cordex_tasmax]
    if rcp == cfg.rcp_ref:
        ds_scen_tasmax["units"] = "C"
        ds_scen_tasmax.attrs["units"] = "C"
```

```
idx_thresh_str_tasmax = idx_threshs_str[0]
arr_idx = indices.tx_days_above(ds_scen_tasmax, idx_thresh_str_tasmax).values
idx_units = 1
```

xarray.DataArray

36



Ajout d'un indice climatique

config.py & launch.py

```
# Activation du calcul des indices climatiques et declaration des indices et seuils.
cfg.opt_idx      = True
cfg.tx_days_above = "tx_days_above"
cfg.tx_tn_days_above = "tx_tn_days_above" ←
cfg.idx_names    = [cfg.tx_days_above, cfg.tx_tn_days_above]
cfg.idx_threshs  = [[36],[30,22]]
```

indices.py

```
# Sélection des variables (CORDEX) nécessaires au calcul des indices climatiques.
vars = []
if idx_name == cfg.idx_tx_days_above:
    vars = [cfg.var_cordex_tasmax]
elif idx_name == cfg.idx_tx_tn_days_above:
    vars = [cfg.var_cordex_tasmax, cfg.var_cordex_tasmin]

# Calcul d'un indice: nombre de jours par an où Tmax > seuil.
ds_idx = None
if idx_name == cfg.idx_tx_days_above:
    ds_scen_tasmax = ds_scen[0][cfg.var_cordex_tasmax]
    if rcp == cfg.rcp_ref:
        ...
        idx_thresh_str_tasmax = idx_threshs_str[0]
        arr_idx = indices.tx_days_above(ds_scen_tasmax, idx_thresh_str_tasmax).values
        idx_units = 1
    # Calcul d'un indice: nombre de jours où Tmax > seuil #1 et Tmin > seuil #2.
elif idx_name == cfg.idx_tx_tn_days_above:
    ds_scen_tasmax = ds_scen[0][cfg.var_cordex_tasmax]
    ds_scen_tasmin = ds_scen[0][cfg.var_cordex_tasmin]
    if rcp == cfg.rcp_ref:
        ...
        idx_thresh_str_tasmax = idx_threshs_str[0]
        idx_thresh_str_tasmin = idx_threshs_str[1]
        arr_idx = indices.tx_tn_days_above(ds_scen_tasmax, ds_scen_tasmin, idx_thresh_str_tasmax,
                                           idx_thresh_str_tasmin).values
        idx_units = 1
```

xarray.DataArray

30

22

Le code en bleu correspond à l'indice climatique ajouté.



The screenshot shows the xclim documentation website. The left sidebar contains a navigation menu with sections like 'CONTENTS', 'Installation', 'Basic Usage', 'Examples', 'Climate indicators', 'Health checks', 'Unit handling', 'Internationalization', 'ICCLIM indices', 'Bias adjustment and downscaling algorithms', 'API', 'Contributing', 'Credits', and 'History'. The main content area displays the 'Number of days with tmax > 30 degc and tmin > 22 degc' indicator, which is highlighted with a red box. The indicator's description and code are shown, with the code snippet in blue matching the code in the indices.py file.

Structure

```
~/Documents/dev/scen_workflow_afr/*.py
/<d_exec>/<country>/<project>/obs/<obs_src>
+---/<variable>/<variable>_<station>.nc;.csv
+---/<liste_stations>.csv
/<d_ra_raw>/<var>/<var>_<obs_src>_hour_<année>/.nc
/<d_ra_day>/<var>/<var>_<obs_src>_day_<année>/.nc
/<d_proj>/<inst>/<rcm>/<gcm>/day/atmos/<version>/<var>/*.nc
/<d_exec>/sim_climat/<country>/<project>/
+---/log/<date>_<time>.log
+---config.ini
+---calib.csv
+---/gis/*.geojson
+---/stn/<stn>/obs/<var>/<var>_<stn>.nc
+---/stn/<stn>/scen/raw/<var>/<var>_<sim>.nc
+---/stn/<stn>/scen/regrid/<var>/<var>_<sim>*.nc (3 fichiers)
+---/stn/<stn>/scen/qgmap/<var>/<var>_<sim>.nc
+---/stn/<stn>/idx/<idx>/<idx>_<sim>.nc
+---/stn/<stn>/stat/<var|idx>/<var|idx>_<stn>.csv
+---/stn/<stn>/fig/
|   +---/calib/<var>/nq_<nq>_upqmf_<upqmf>_timewin_<timewin>/<var>_<sim>_calib*.png
|   +---/postprocess/<var>/<var>_<sim>_postprocess.png
|   +---/wflow/<var>/<var>_<sim>_wflow.png
|   +---/idx/<idx>_<station>.png
+---/fig/idx/<idx>_<rcp>_<hor>*.png

# Code source Python.
# Données météorologiques observées.
# Équivalent dans un format NetCDF.
# Liste des stations.
# Réanalyses (fréquence originale/horaire).
# Réanalyses (fréquence journalière).
# Simulations climatiques (fréquence journalière).
# Sortie de l'analyse.
# Fichier de log.
# Fichier de paramètres de configuration.
# Fichier de calibration (export).
# Polygone représentant le contour du pays.
# Observations.
# Scénarios (après extraction spatiotemporelle).
# Scénarios (après ajustement de la grille).
# Scénarios (après quantile mapping).
# Indices climatiques.
# Statistiques d'ensemble (export).
# Figures.
# Ajustement de biais.
# Post-traitement.
# Flux de travail.
# Indices climatiques (séries temporelles).
# Indices climatiques (cartes).
```

<d_exec> : répertoire des résultats

<d_ra_raw> : répertoire des réanalyses (fréquence originale)

<stn> : station météorologique (si observations disponibles)

<var> : variable (selon la toponymie de CORDEX)

<obs_src> : source des observations (ex: acronyme de l'institut, "era5", "era5_land")

<sim> : simulation climatique = <rcm>_<gcm>_<rcp>

<rcm> : modèle climatique régional

<d_raw_day> : répertoire de réanalyses (fréquence journalière)

<idx> : indice climatique

<hor> : horizon climatique

<inst> : institut (simulation)

<gcm> : modèle climatique global

<version> : version

<rcp> : scén. émissions GES

<project> : projet

<country> : pays

ANALYSE DE L'INFORMATION CLIMATIQUE

4. ENVIRONNEMENT DE DÉVELOPPEMENT



python - interpréteur

Terminal

```
$ sudo apt update  
$ sudo apt install python  
$ sudo apt install python-pip  
$ sudo apt install python3  
$ sudo apt install python3-pip
```

anaconda (incl. numpy et pandas) - distribution Python pour programmation scientifique

Terminal

```
# Installer:
$ curl -O https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86_64.sh
$ sha256sum Anaconda3-2020.02-Linux-x86_64.sh
$ bash Anaconda3-2020.02-Linux-x86_64.sh

# Configurer:
$ "PATH=~/.anaconda3/bin:$PATH" >> ~/.bashrc
$ source ~/.bashrc
$ conda info
$ conda update conda

# Créer un environnement de développement:
$ conda create --name xclim-dev python=3.7.6

# Fermer et ouvrir un nouveau terminal:
$ conda init bash

# Ajouter "-c conda-forge" à la commande "conda install":
$ conda activate base
$ conda config --add channels conda-forge

# Charger le module Anaconda:
$ module avail
$ module load Anaconda
$ source activate python37
```

xclim - librairie Python pour le calcul d'indices climatiques

Terminal

```
# Installation à l'aide d'Anaconda:
$ conda activate xclim-dev
$ conda install -c conda-forge xclim

# Modification de xclim:
$ cd ~/Documents/dev/
$ git clone https://github.com/yrouranos/xclim.git
$ cd xclim

# Créer un environnement Anaconda:
$ conda env create --file environment.yml --name xclim-dev

# Supprimer un environnement Anaconda (si nécessaire):
$ conda activate xclim-dev

# Supprimer un environnement (non requis):
$ conda env remove --name xclim-dev

# Mise à jour (version stable):
$ pip install -e .
$ conda update xclim

# Pour utiliser tout ça dans pycharm:
$ conda activate xclim-dev
$ conda install jupyter
```


xclim - librairie Python pour le calcul d'indices climatiques

Terminal

```
# Pour créer un environnement Python:
$ python -m venv

# Vérifier que tout a bien fonctionné (dans l'environnement approprié).
# Ça retourne une erreur si ça ne fonctionne pas.
$ python -c "import xclim"

# Mettre à jour les librairies comprises dans l'environnement:
$ conda env update -n xclim-dev -f environment.yml
$ pip install -e .

# Mettre à jour pip:
$ pip install --no-cache-dir

# Installer des librairies (dans le terminal de PyCharm):
$ pip install cdsapi
$ pip install boltons
$ pip install pytest-runner
$ pip install tox
```

téléchargement
automatisé des
données ERA5 et
ERA5-Land

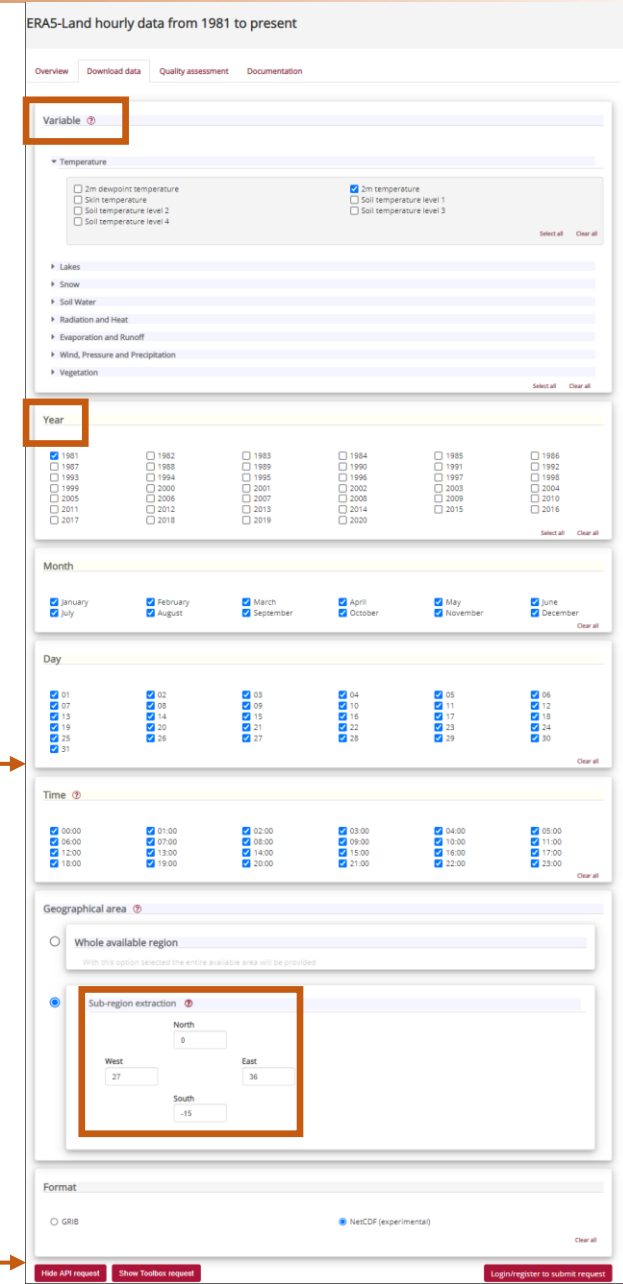
Téléchargement des données ERA5 et/ou ERA5-Land

- Création d'un compte Copernicus
<https://cds.climate.copernicus.eu/user/register?destination=%2Fcdsapp%23!%2Fhome>
- Identification des informations de sécurité
<https://cds.climate.copernicus.eu/api-how-to>
Connexion au portail
Copie du UID et de l'API Key (confidentiel) dans un fichier ~/.cdsapirc

```
~/.cdsapirc
```

```
url: https://cds.climate.copernicus.eu/api/v2  
key: <UID>:<API Key>
```

- Identification des données d'intérêt
<https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-land?tab=form>
Le code Python de téléchargement est ici
- Identifier le nom et le code des variables pertinentes
Ex: le code de la variable "2m_temperature" est "t2m".
Spécifier les variables requises dans config.ini.
- Exécuter le script avec l'option `opt_download = True`



The screenshot shows the 'ERA5-Land hourly data from 1981 to present' selection interface. The 'Variable' section is highlighted with a red box, showing '2m temperature' selected. The 'Year' section is also highlighted with a red box, showing years from 1981 to 2020. The 'Month' section shows all months selected. The 'Day' section shows all days selected. The 'Time' section shows all times selected. The 'Geographical area' section is highlighted with a red box, showing 'Sub-region extraction' with coordinates: North 0, West 27, East 36, South -15. The 'Format' section shows 'NetCDF (experimental)' selected. At the bottom, there are buttons for 'Hide API request', 'Show Toolbox request', and 'Login/register to submit request'.

pycharm - Integrated Development Environment (Python)

Terminal

```
# Installer pycharm:
$ sudo app install snap
$ sudo snap install pycharm-community --classic

# Importer des librairies à partir du terminal:
$ conda activate xclim-dev
$ sudo apt-get install gdal-bin
$ sudo apt-get install libgdal-dev libgdal1h
$ pip install GDAL==$(gdal-config --version) --global-option=build_ext --global-option="-I/usr/include/gdal"

# Importer des librairies à partir de l'éditeur (si non reconnu):
cartopy
xclim
```

idv (Unidata)

Terminal

```
# Télécharger:
$ curl -O https://www.unidata.ucar.edu/downloads/idv/current/index.jsp

# Installer:
$ bash idv_5_7_linux64_installer.sh

# Lancer:
$ ./runIDV
```

panoply (NASA)

Terminal

```
# Télécharger:
$ curl -O https://www.giss.nasa.gov/tools/panoply/download/PanoplyJ-4.11.5.zip

# Installer:
$ cd ~/Documents/prog/panoplyJ-4.11.5/
$ chmod guo+x panoply.sh

# Lancer:
$ ./panoply.sh
```

ANALYSE DE L'INFORMATION CLIMATIQUE

5. EXÉCUTION DU SCRIPT



Comportement général lors de la génération des résultats

- Fichiers.nc (données) non remplacés s'ils existent déjà
Permet de poursuivre le traitement qui aurait arrêté (ex: panne électrique, anomalie).
- Fichiers .png (diagrammes et cartes) remplacés
Permet de régénérer les éléments visuels sans tout recalculer
- Répertoires créés automatiquement

Lancement du script

- Saisir les options dans config.ini et launch.py.
- Lancement dans PyCharm ou à partir d'un terminal.
- Script plus stable dans un terminal.

Terminal

```
$ python3 launch.py
```

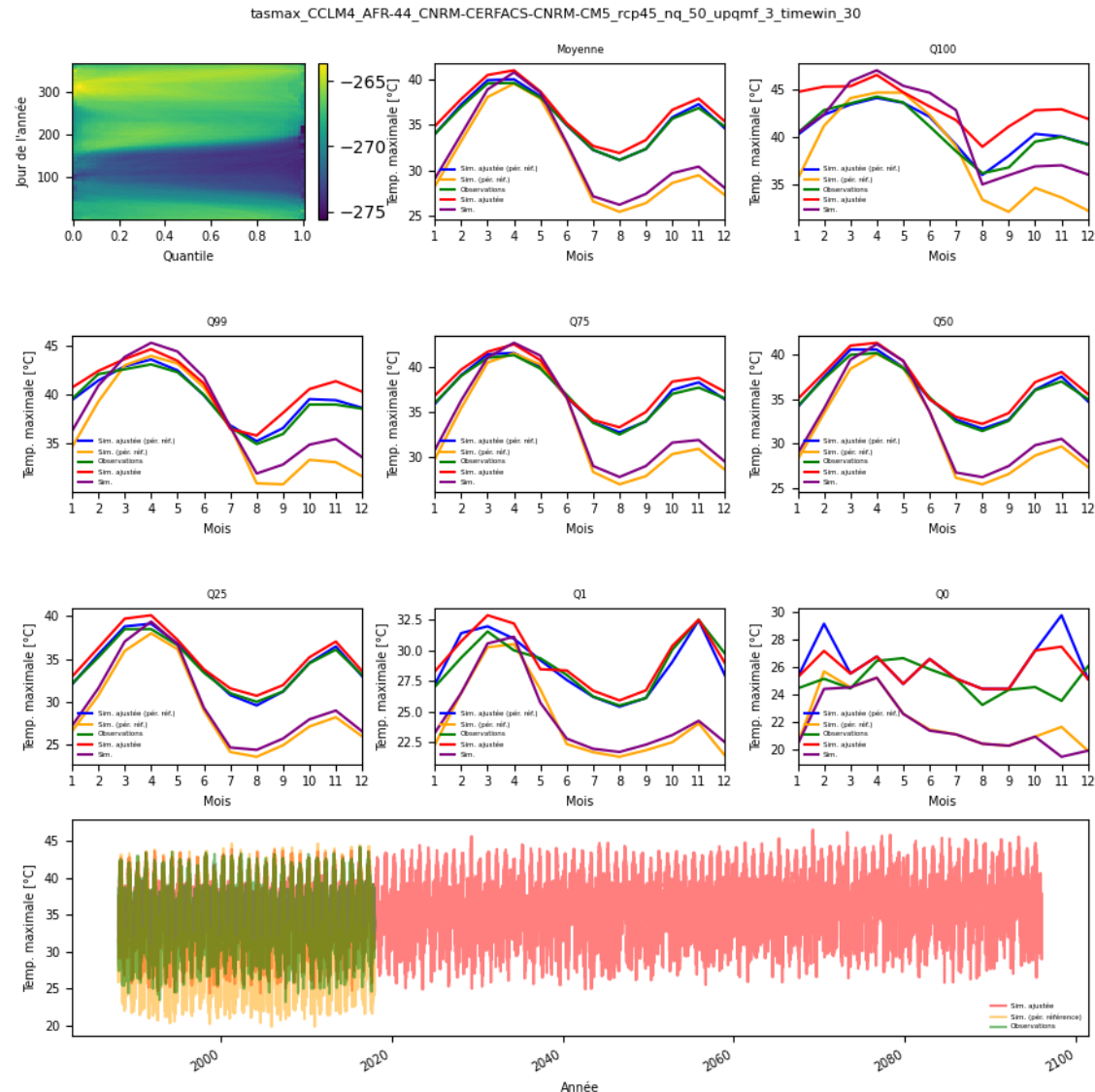
Performance

- Traitement en parallèle.
- Vitesse de lecture/écriture.
- Volume important des fichiers de sortie.

Résolution d'une anomalie

- Regarder dans le fichier .log pour des indices.
- Ajouter des 'breakpoints' dans PyCharm, sélectionner options adéquates, et lancer en mode 'debug'.

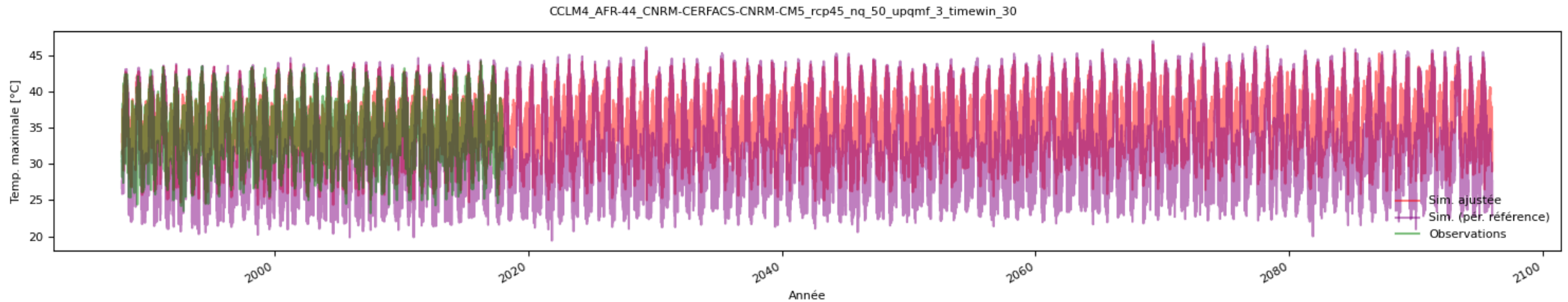

```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/calib/<var>/nq_<nq>_upqmf_<upqmf>_timewin_<timewin>/<var>_<sim>_calib.png
```



Notes

- Vérification pour chacune des combinaisons de station - simulation - variable - scénario d'émission - paramètres d'ajustement de biais et de mise à l'échelle statistique.
- Courbes de valeurs moyennes mensuelles pour quelques quantiles (100, 99, 75, 50, 25, 1 et 0) et pour la moyenne.
- Série temporelle.
- Différence importante entre le résultat d'une simulation climatique et les observations pour les températures basses (voir la série temporelle).
- Meilleure correspondance entre la simulation ajustée et les observations suite à l'ajustement du biais et la mise à l'échelle statistique.

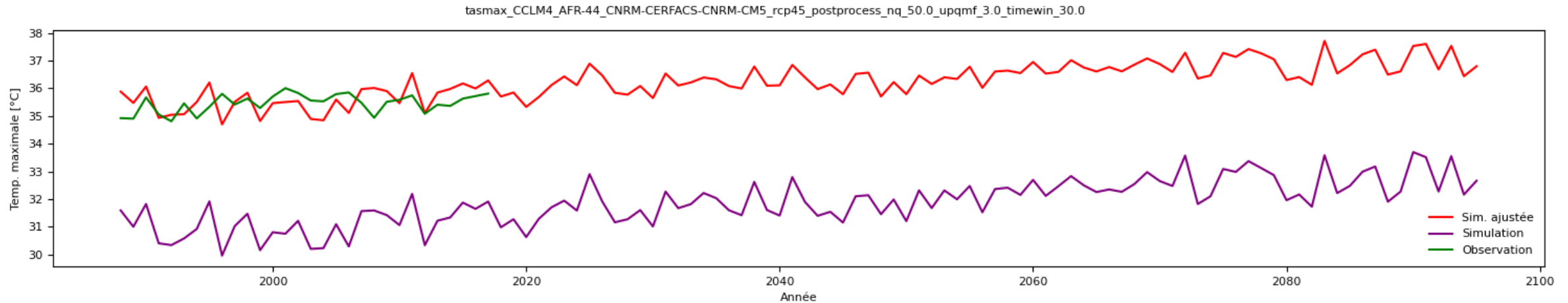
```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/calib/<var>/nq_<nq>_upqmf_<upqmf>_timewin_<timewin>/<var>_<sim>_calib_ts.png
```



Notes

- Il s'agit de la série temporelle qui a été présentée à la page précédente, en plus gros.
- Il y a une différence importante entre le résultat de la simulation climatique et les observations pour les températures basses.

```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/postproces/<var>/<var>_<sim>_postprocess*.png
```

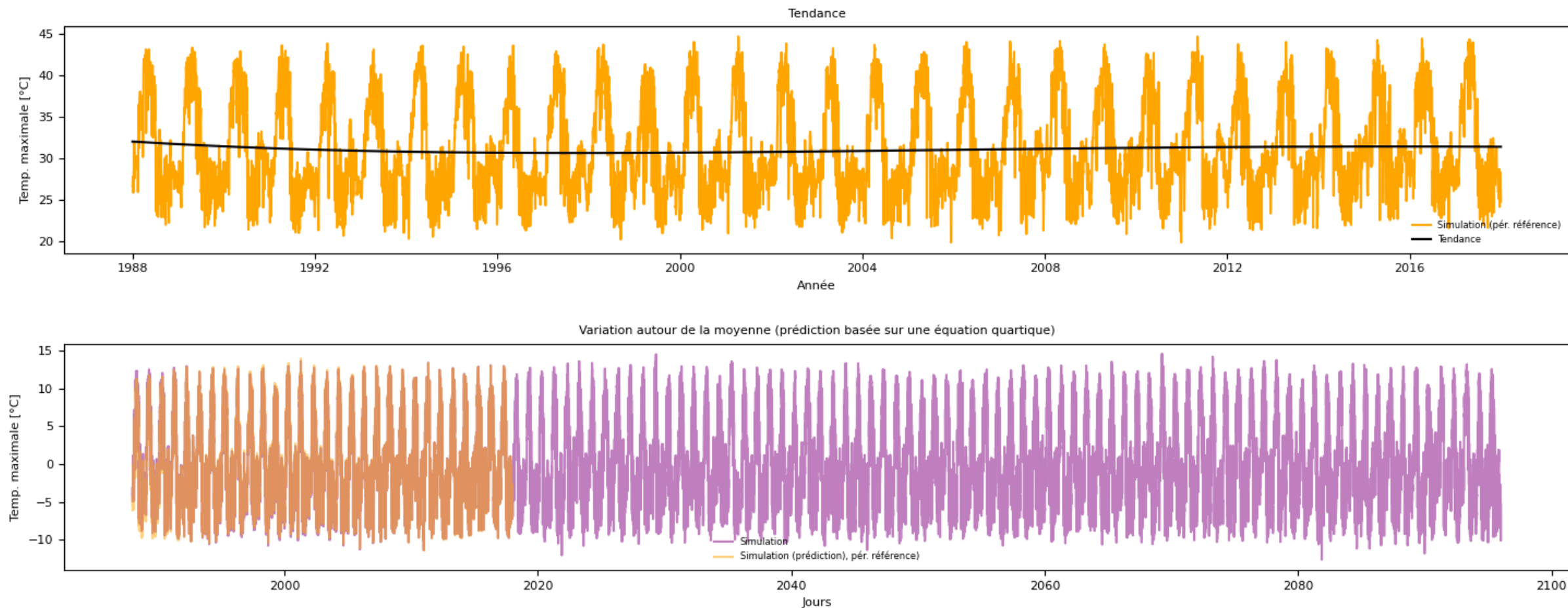


Notes

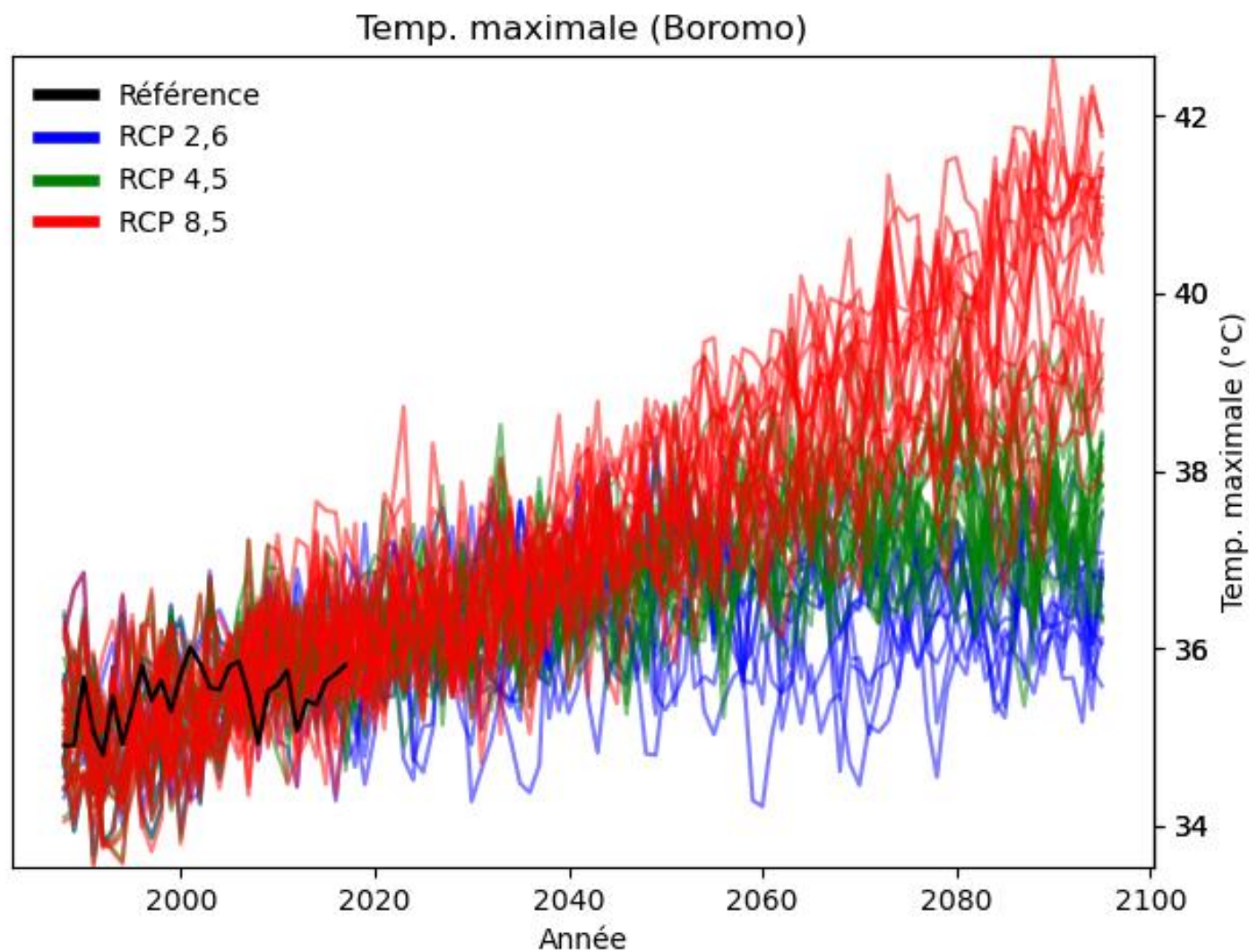
- L'idée est la même qu'à la page précédente mais il s'agit de valeurs annuelles (au lieu de journalières).
- La simulation climatique donne des valeurs qui sont environ 3°C plus basses que les observations météorologiques enregistrées à la station.
- L'ajustement de biais et la mise à l'échelle statistiques sont des étapes essentielles.

```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/wflow/<var>/<var>_<sim>_wflow*.png
```

tasmax_CCLM4_AFR-44_CNRM-CERFACS-CNRM-CM5_rcp45_workflow_nq_50_upqmf_3.0_timewin_30



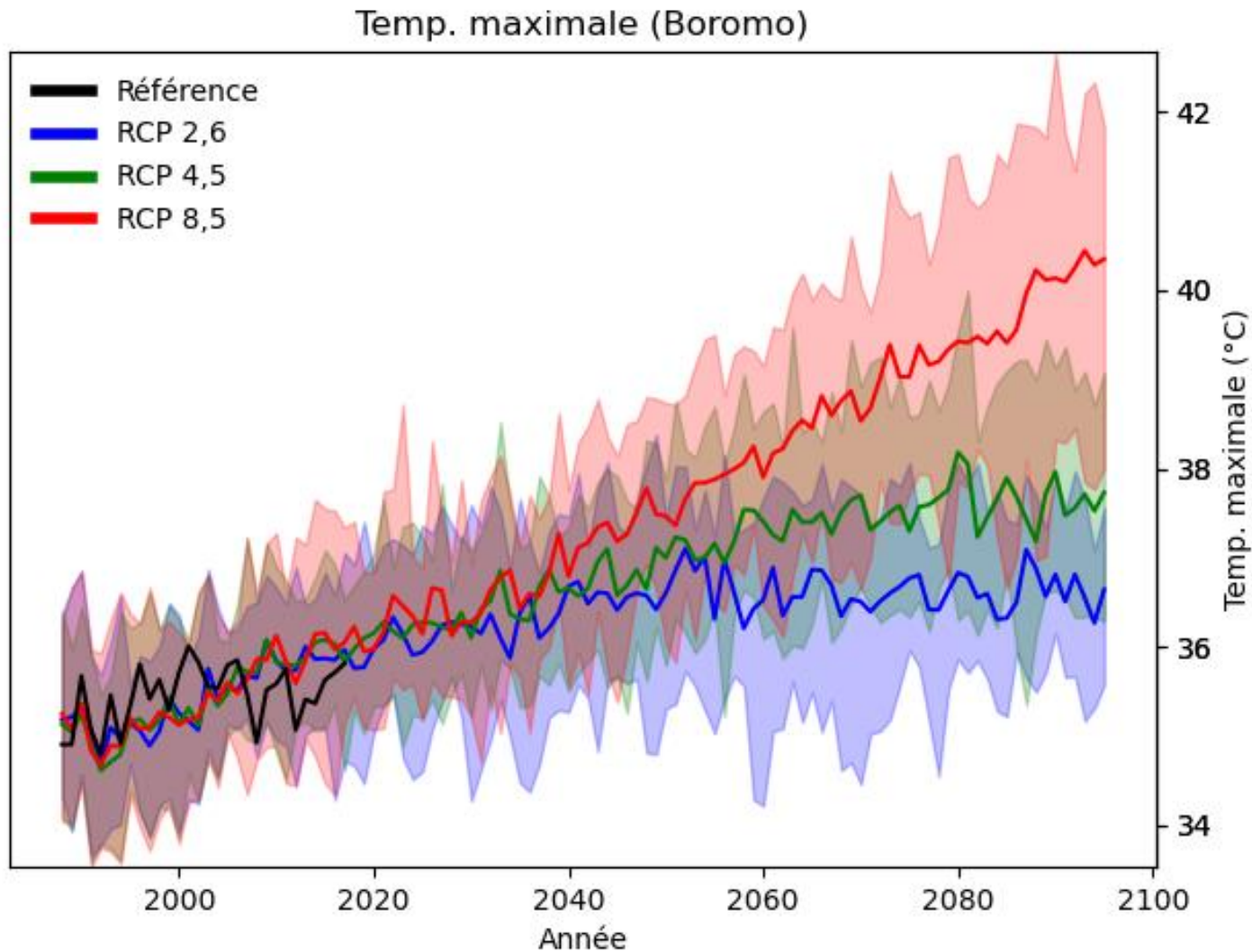
```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/scen/<var>_<station>_sim.png
```



Notes

- Chacune des courbes est associée à la sortie d'une simulation.
- Ce type de diagramme sera utilisé lors des analyses de vulnérabilités et d'impacts.

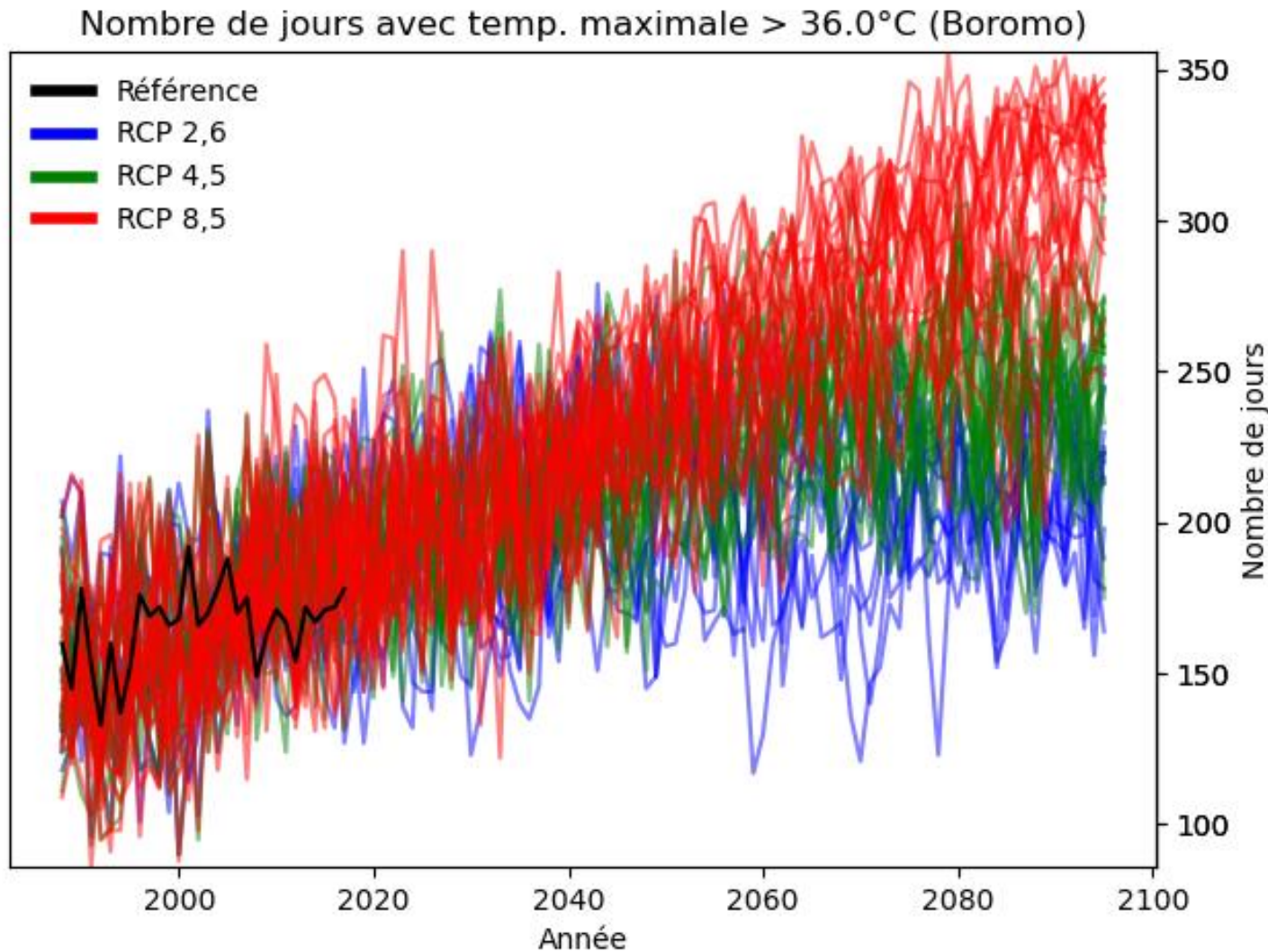

```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/scen/<var>_<station>_rcp.png
```



Notes

- Les zones ombragées correspondent à l'ensemble des simulations appartenant à une catégorie de scénarios d'émissions de GES.
- Ce type de diagramme sera utilisé lors des analyses de vulnérabilités et d'impacts.

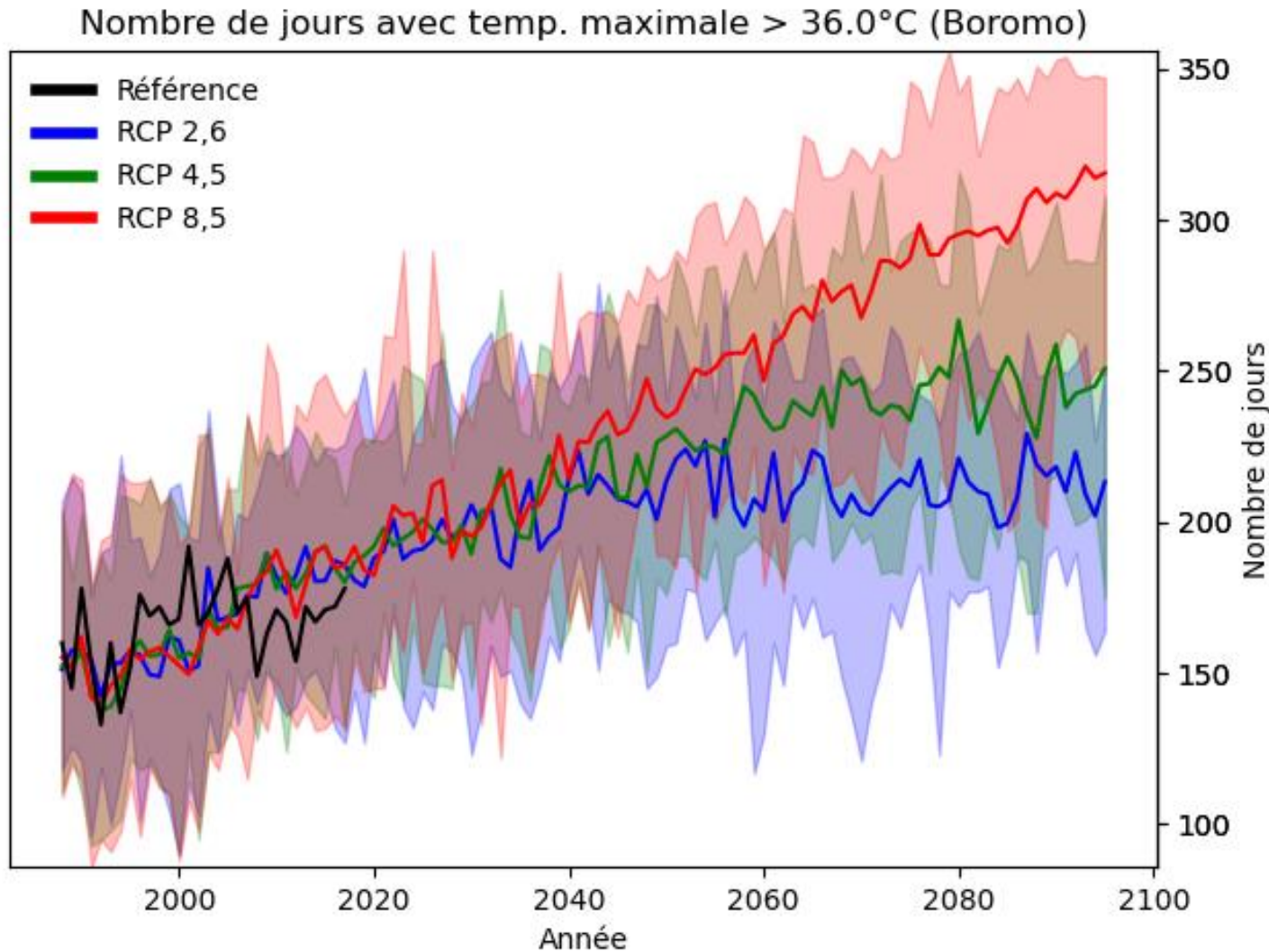
```
/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/idx/<idx>_<station>_rcp.png
```



Notes

- Chacune des courbes est associée à la sortie d'une simulation
- Ce type de diagramme sera utilisé lors des analyses de vulnérabilités et d'impacts.

/exec/<username>/sim_climat/<country>/<project>/<stn>/fig/idx/<idx>_<station>_sim.png

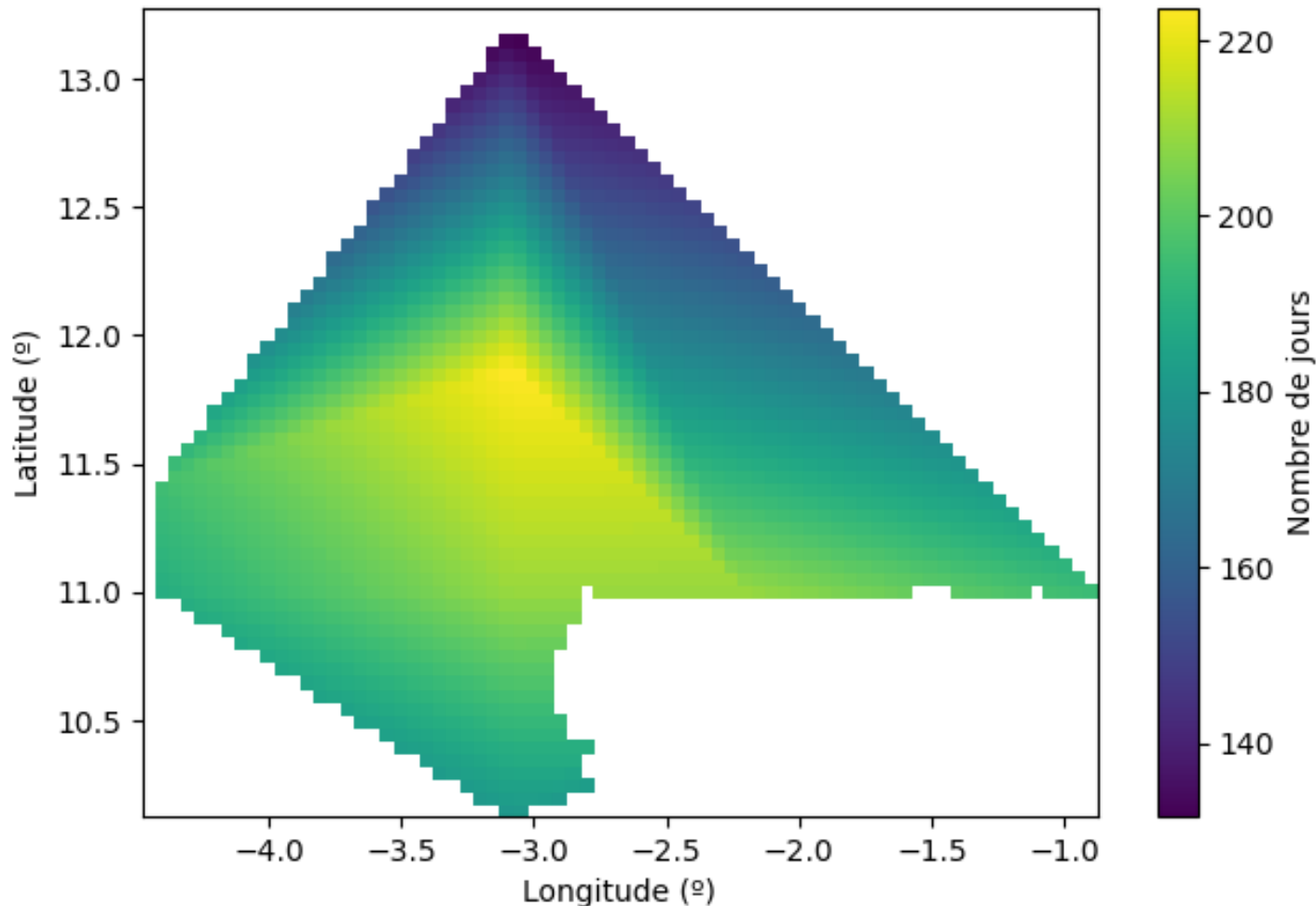


Notes

- Les zones ombragées correspondent à l'ensemble des simulations appartenant à une catégorie de scénarios d'émissions de GES.
- Revoir la section sur les concepts en scénarisation climatique pour les explications à propos de cette figure.
- Ce type de diagramme sera utilisé lors des analyses de vulnérabilités et d'impacts.

```
/exec/<username>/sim_climat/<country>/<project>/fig/indices/<idx>_<rcp>_<horizon>.png
```

Nombre de jours avec temp. maximale > 36.0 °C (Burkina, 2041-2060)



Notes

- Apparence discutable étant donné le faible nombre de stations pour la variable 'tasmax' (qui a servi à calculer l'indice) et le jeu de données incomplet de la station au nord.
- Il faut établir des règles pour déterminer si un jeu de données est acceptable et l'exclure si les données sont insuffisantes.
- Les stations ne couvrent pas la superficie du Burkina Faso.
- Une interpolation dans un système d'information géographique (SIG) donnerait de meilleurs résultats.

Fichier .log

- Documentation de l'analyse qui permet de suivre le progrès et de trouver l'emplacement d'une anomalie.
- Génération automatique du fichier lors du lancement du script.

Ex: 20200805_132537.log indique que le script a été lancé le 5 août 2020 à 13h25 et 37 secondes.

```

/exec/<username>/sim_climat/<country>/<project>/log/<date>_<time>.log

=====
PRODUCTION OF CLIMATE SCENARIOS & CALCULATION OF CLIMATE INDICES
Python Script created by Ouranos, based on xclim and xarray libraries.
=====
Country      : burkina
Project      : pcci
CORDEX variables : ['tas', 'tasmin', 'tasmax', 'pr', 'uas', 'vas']
Climate index #1 : tx_days_above[36.0]
Stations     : ['bereba', 'boromo', 'boura', 'diebougou', 'farakoba',
                'gao', 'gaoua', 'hounde', 'kassoum', 'koumbia', 'leo',
                'nasso', 'po', 'sapouy', 'valleedukou']
Emission scenarios : ['rcp26', 'rcp45', 'rcp85']
Reference period  : [1988, 2017]
Future period    : [1988, 2095]
Horizons        : [[2021, 2040], [2041, 2060], [2061, 2080]]
=====
Step #2a Downloading climate data is not required
=====
Step #2b Aggregation of hourly data is not required
        Calibration file loaded.
=====
Step #2c Converting observations from CSV to NetCDF files
=====
Step #2d Listing directories with CORDEX files
=====
Variable : tas
Station  : boromo
RCP      : rcp26
Simulation : REMO2009_AFR-44_MIROC-MIROC5_rcp26

=====
Step #3-4 Spatial & temporal extraction and grid transfer (or interpolation) is
        not required
=====
Step #4.5 Pre-processing is not required
=====
Step #5a Calculating adjustment factors is running
        Assessing REMO2009_AFR-44_MIROC-MIROC5_rcp26: nq=50, up_qmf=3, time_win=30
        Selected parameters: nq=50.0, up_qmf=3.0, time_win=30.0, bias_err=-1.0
=====
Step #5bc Statistical downscaling and bias adjustment is not required
=====
Step #6 Calculation of climate indices is running.
Step #6a Calculation of indices.
=====
Index      : tx_days_above
Station    : boromo
Emission scenario : rcp26
=====
Collecting simulation files.
Calculating climate indices.
Generating NetCDF files containing indices.
Generating time series of indices.
=====
Step #6b Generation of index maps.
        Collecting emissions scenarios at all stations.
=====
Step #7a Calculation of statistics for climate scenarios.
        Processing: station = boromo; variable = tas
=====
Step #7b Calculation of statistics for climate indices.
        Processing: station = boromo; index = tx_days_above
Script completed successfully.

```

Fichier calib.csv

- Paramètres pour l'ajustement du biais et la mise à l'échelle statistique (nq, up_qmf, time_win).
- Chaque ligne du fichier correspond à la combinaison de paramètres qui minimise l'erreur (différence) entre la simulation ajustée et les observations pour la période de référence.
- Un ensemble de paramètres par combinaison de station, simulation et variable. Quelques milliers de combinaisons.
- Génération automatique du fichier lors de l'exécution du script.
- Tentative de lecture du fichier pour utiliser les combinaisons spécifiées manuellement ou celles déterminées automatiquement lors de l'exécution précédente du script.

```
/exec/<username>/sim_climat/<country>/<project>/calib.csv
```

```
id,sim_name,stn,var,nq,up_qmf,time_win,bias_err
0,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,bereba,tas,50,3,30,-1
1,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,bereba,tasmin,50,3,30,-1
2,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,bereba,tasmax,50,3,30,-1
3,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,bereba,pr,50,3,30,-1
4,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,bereba,uas,50,3,30,-1
5,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,bereba,vas,50,3,30,-1
6,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boromo,tas,50,3,30,-1
7,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boromo,tasmin,50,3,30,-1
8,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boromo,tasmax,50,3,30,-1
9,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boromo,pr,50,3,30,-1
10,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boromo,uas,50,3,30,-1
11,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boromo,vas,50,3,30,-1
12,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boura,tas,50,3,30,-1
13,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boura,tasmin,50,3,30,-1
14,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boura,tasmax,50,3,30,-1
15,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boura,pr,50,3,30,-1
16,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boura,uas,50,3,30,-1
17,CSC_AFR-44_MPI-M-MPI-ESM-LR_rcp26,boura,vas,50,3,30,-1
```

Fichier de statistiques

- Variables climatiques (ex: `tasmax`) et indices climatiques (ex: `tx_days_above`)
- Observations aux stations (`ref`) et scénarios climatiques (`rcp`)
- Valeurs moyenne (`mean`), minimum (`min`), maximum (`max`) et quantiles (`q`)
- Les quantiles sont définis dans `config.py`: `stat_quantiles = [1.00, 0.99, 0.75, 0.50, 0.25, 0.01, 0.00]`

```
/exec/<username>/sim_climat/<country>/<project>/stn/<stn>/stat/<var|idx>/<var/idx>_<stn>.csv
```

```
id,stn,var,rcp,hor,stat,q,val
0,boromo,tx_days_above,ref,1988-2017,none,-1,165.733333
1,boromo,tx_days_above,rcp26,2021-2040,mean,-1,197.346154
2,boromo,tx_days_above,rcp26,2041-2060,mean,-1,212.007692
3,boromo,tx_days_above,rcp26,2061-2080,mean,-1,210.992308
4,boromo,tx_days_above,rcp26,2021-2040,min,-1,148.6
5,boromo,tx_days_above,rcp26,2041-2060,min,-1,161.85
6,boromo,tx_days_above,rcp26,2061-2080,min,-1,159.05
7,boromo,tx_days_above,rcp26,2021-2040,max,-1,243.7
8,boromo,tx_days_above,rcp26,2041-2060,max,-1,254.2
9,boromo,tx_days_above,rcp26,2061-2080,max,-1,251.8
10,boromo,tx_days_above,rcp26,2021-2040,quantile,0.99,242.332
11,boromo,tx_days_above,rcp26,2041-2060,quantile,0.99,252.874
12,boromo,tx_days_above,rcp26,2061-2080,quantile,0.99,250.72
13,boromo,tx_days_above,rcp26,2021-2040,quantile,0.75,214.35
14,boromo,tx_days_above,rcp26,2041-2060,quantile,0.75,230.05
15,boromo,tx_days_above,rcp26,2061-2080,quantile,0.75,230.8
16,boromo,tx_days_above,rcp26,2021-2040,quantile,0.5,196.45
17,boromo,tx_days_above,rcp26,2041-2060,quantile,0.5,214.6
18,boromo,tx_days_above,rcp26,2061-2080,quantile,0.5,213.05
19,boromo,tx_days_above,rcp26,2021-2040,quantile,0.25,178.75
20,boromo,tx_days_above,rcp26,2041-2060,quantile,0.25,195.65
21,boromo,tx_days_above,rcp26,2061-2080,quantile,0.25,193.95
22,boromo,tx_days_above,rcp26,2021-2040,quantile,0.01,150.538
23,boromo,tx_days_above,rcp26,2041-2060,quantile,0.01,163.584
24,boromo,tx_days_above,rcp26,2061-2080,quantile,0.01,160.928
25,boromo,tx_days_above,rcp45,2021-2040,mean,-1,201.4875
```

Indice climatique:
`tx_days_above`

ANALYSE DE L'INFORMATION CLIMATIQUE

6. PROCHAINES ÉTAPES



1. Identification d'indices climatiques pertinents (et des seuils associés)
 - Tenir compte de la littérature scientifique et du savoir local.
2. Bonification du code Python (facultatif)
 - Intégrer des indices climatiques additionnels.
 - Diversifier les éléments visuels.
3. Production des scénarios climatiques et calcul d'indices climatiques
 - Fichiers de données (*.nc) et conversion vers SIG.
 - Éléments visuels (diagramme et cartes) (*.png)
 - Statistiques (*.csv).
4. Utilisation des résultats
 - Interpréter les sorties des scénarios climatiques.
 - Identifier les vulnérabilités en égard d'une activité socio-économique.
 - Anticiper les impacts du changement climatique.
 - Élaborer des stratégies d'adaptation.
 - Estimer les coûts et bénéfices.
 - Diffuser l'information aux usagers de l'information climatique (ex: rapports, site web, Atlas)