Subcategory. Climate Data Analysis in RStudio

Dmytro Diadin, O. M. Beketov National University of Urban Economy in Kharkiv

About the Resource

This resource is part of the Climate Data and Tools section of the ClimEd Scientific and Educational Platform. It is intended for students, PhD candidates, teachers, researchers, and professionals working in climate-sensitive sectors who use climate data, spatial datasets, time series, reanalysis products, climate scenarios, and projections.

The material explains how to use R and RStudio for automated data retrieval, preprocessing, spatial analysis, aggregation, calculation of climate indicators, comparison of periods and scenarios, and preparation of tables, maps, and graphs. The resource can be used in ClimEd courses, practical and laboratory assignments, qualification and research projects, climate risk assessments, and the development of applied climate services.

Key Message

R and RStudio should be used to automate repetitive operations and conduct reproducible analyses of large climate datasets, while QGIS or ArcGIS can be used at the final stage to verify, map, and visualise the results.

Subcategories for subcategory “Climate Data Analysis in RStudio”

1. Why Use R and RStudio for Climate Data
2. Typical Automated Processing Workflow
3. Main R Packages for Climate and Spatial Data
4. Working with Time Series and Climate Indicators
5. Comparing Climate Periods, Models, and Scenarios
6. Example Structure of an R Script
7. Combining Climate Data with Territorial Boundaries
8. Calculating Composite Climate Risk Indicators
9. Comparing Manual GIS Workflows and Automation in R
10. Practical Recommendations and Good Practice
11. Using the Resource in Education and Research

1. Why Use R and RStudio for Climate Data

R and RStudio provide a convenient environment for automated work with climate data, particularly when large collections of NetCDF (.nc) files need to be processed without first opening them in QGIS or ArcGIS. Unlike manual GIS workflows, R makes it possible to describe the entire processing sequence in a script: loading files, selecting a variable, clipping data to a study area, aggregating data over time, calculating indicators, saving results, and producing graphs. This substantially accelerates analysis, makes it reproducible, and reduces the risk of errors when working with dozens or hundreds of climate layers.

> **Terminology Note**
>
> R is a programming language and environment for statistical computing. RStudio is an integrated development environment that facilitates writing, running, and documenting R scripts. In practice, they are usually used together.

2. Typical Automated Processing Workflow

A typical workflow includes preparing the source files, loading the required packages, opening the data, selecting a climate variable, defining the temporal and spatial extent, aggregating the data, calculating indicators, checking the results, and exporting the outputs.

1. Load the packages required for working with raster, vector, and tabular data.
2. Open a file or a series of files in NetCDF, GeoTIFF, CSV, or another supported format.
3. Select the required variable, units of measurement, time range, and study area.
4. Check the coordinate reference system, time dimension, missing values, and metadata.
5. Perform spatial cropping, masking, reprojection, or resampling.
6. Aggregate the data by month, season, year, or other periods.
7. Calculate climate indicators, statistics, anomalies, or changes.
8. Export the results and document the analysis parameters.

Figure 1 – Climate data processing workflow in R

3. Main R Packages for Climate and Spatial Data

The packages most commonly used for working with climate rasters in R include terra, stars, raster, ncdf4, sf, exactextractr, tidyverse, and lubridate. They can open NetCDF files directly, select the required climate variable, work with time dimensions, crop data to the boundaries of a region, community, or river basin, reproject data, change spatial resolution, mask values using a polygon, and export results to GeoTIFF, NetCDF, CSV, or Excel. R can therefore perform the same basic operations as desktop GIS software, but in batch mode and for many files simultaneously.

Package	Main Purpose	Typical Tasks
terra	Modern processing of raster and	Opening NetCDF/GeoTIFF, cropping,

	vector data	masking, aggregation, and export
stars	Multidimensional spatiotemporal arrays	Working with data cubes and time dimensions
ncdf4	Low-level work with NetCDF	Reading variables, dimensions, attributes, and metadata
sf	Vector geospatial data	Community, regional, and basin boundaries; reprojection and spatial operations
exactextractr	Exact zonal statistics	Raster means, sums, and other statistics within polygons
tidyverse	Tabular processing and visualisation	Cleaning, transformation, grouping, and preparation of tables and graphs
lubridate	Working with dates and time	Defining seasons, years, months, and time intervals

Good Practice

For new projects, the terra package should generally be preferred to the older raster package where the required functionality and workflow compatibility allow. The versions of R, RStudio, and the packages should be recorded in the methodological description or project environment.

4. Working with Time Series and Climate Indicators

R is particularly useful for processing climate time series. For example, daily temperature values can be used to calculate monthly, seasonal, or annual means automatically, as well as the number of hot days, tropical nights, and frost days, the duration of heatwaves, accumulated active temperatures, or the number of days exceeding a specified threshold. For precipitation, it is possible to calculate annual totals, maximum one-day precipitation, maximum five-day precipitation, the number of days with heavy precipitation, the duration of dry spells, drought indices, or percentage changes in precipitation relative to a baseline period.

Examples of Temperature Indicators

- monthly, seasonal, and annual mean temperatures;
- number of hot days and tropical nights;
- number of frost days;
- duration and frequency of heatwaves;
- accumulated active or effective temperatures;
- number of days exceeding specified temperature thresholds.

Examples of Precipitation Indicators

- monthly, seasonal, and annual precipitation totals;
- maximum one-day precipitation;
- maximum five-day precipitation total;
- number of days with heavy or very heavy precipitation;
- duration of dry spells;
- drought indices and relative changes in precipitation.

Important

Before calculating indicators, check the calendar, units of measurement, missing values, completeness of years, and aggregation rules. To ensure reproducibility, retain the definition of each indicator, the threshold values, and the baseline period.

5. Comparing Climate Periods, Models, and Scenarios

R is also well suited to comparing climate periods and scenarios. For example, it can automatically calculate the mean value of an indicator for the 1991–2020 baseline period and separately for the 2021–2040 projection period, and then produce a raster of absolute or relative change. Different greenhouse-gas-emission scenarios, climate models, or ensemble means can be compared in the same way. This is particularly important when working with EURO-CORDEX, CMIP, ERA5-Land, or E-OBS data, where the number of files and time slices can be very large.

- absolute change: future value minus the baseline-period value;
- relative change: the difference expressed as a percentage of the baseline-period value;
- ensemble mean: the mean value across several climate models;
- ensemble range or spread: a measure of inter-model uncertainty;
- scenario comparison: assessment of how results depend on future emission pathways.

6. Example Structure of an R Script

A simple R script for working with climate data may include loading packages, opening a file, selecting a variable, defining a time period, calculating an indicator, and saving the result. Figure 2 illustrates the logic of such a script.

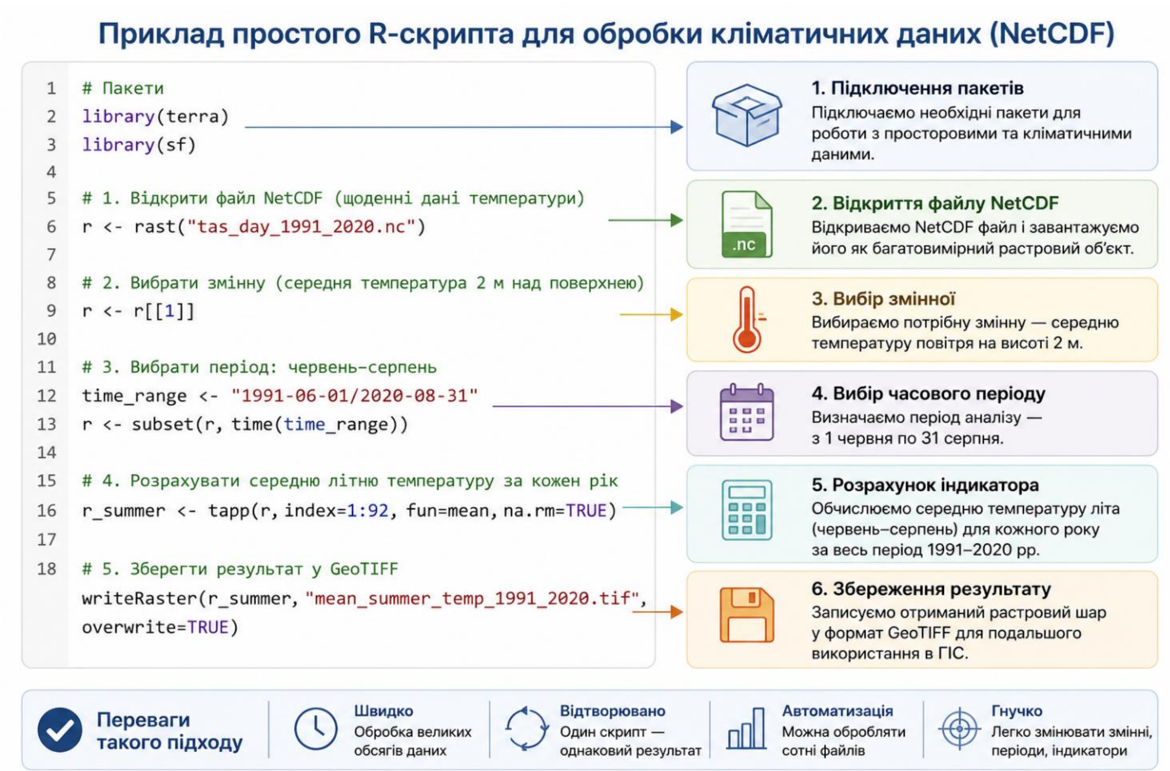

Figure 2 – Example of an R script with explanations of the main operations

Recommendation for Code Presentation

For a website or learning course, code should be provided not only as an image but also as a text block that users can copy. The code should be accompanied by a description of the input-file structure, variable names, units of measurement, and expected output.

7. Combining Climate Data with Territorial Boundaries

Another important advantage of R is its ability to combine climate data with vector boundaries. For example, it can automatically calculate the mean temperature, precipitation, or any climate indicator for each community, region, catchment, or protected area. The result may be a table containing, for each territorial unit, the value of the climate indicator, its projected future change, a hazard class, or a composite index. Such tables can then be readily used in Excel, QGIS, or statistical models.

Spatial Consistency Check

Before calculating zonal statistics, check the coordinate reference systems, polygon geometry, raster spatial resolution, rules for processing partially overlapping cells, and NoData values.

8. Calculating Composite Climate Risk Indicators

In addition to simple calculations, R can be used to create more complex climate risk indicators. For example, several climate indicators can be normalised to a 1–5 scale, assigned weights, and combined into a composite climate hazard index; sensitivity, adaptive capacity, or vulnerability can also be calculated. All these operations can be performed using a consistent methodology across multiple sectors, scenarios, and time horizons, substantially improving the consistency of the results.

- normalisation of indicators to a common scale;
- reclassification using defined thresholds;
- application of weighting coefficients;
- combination of hazard, exposure, sensitivity, and adaptive-capacity indicators;
- sensitivity analysis of results to weights and thresholds;
- preparation of tables and maps of composite risk or vulnerability.

Interpretation

A composite index is not a direct measurement of risk. Its value depends on the selected indicators, normalisation methods, weights, and aggregation rules. The methodology should be transparent, justified, and tested through sensitivity analysis.

9. Comparing Manual GIS Workflows and Automation in R

Using R can substantially accelerate climate data processing by automating repetitive operations and removing the need to open every file in GIS software. When large NetCDF files containing daily climate indicators over several decades need to be processed, R is generally more efficient because it supports batch processing, memory management, and the automatic repetition of identical operations across many files.

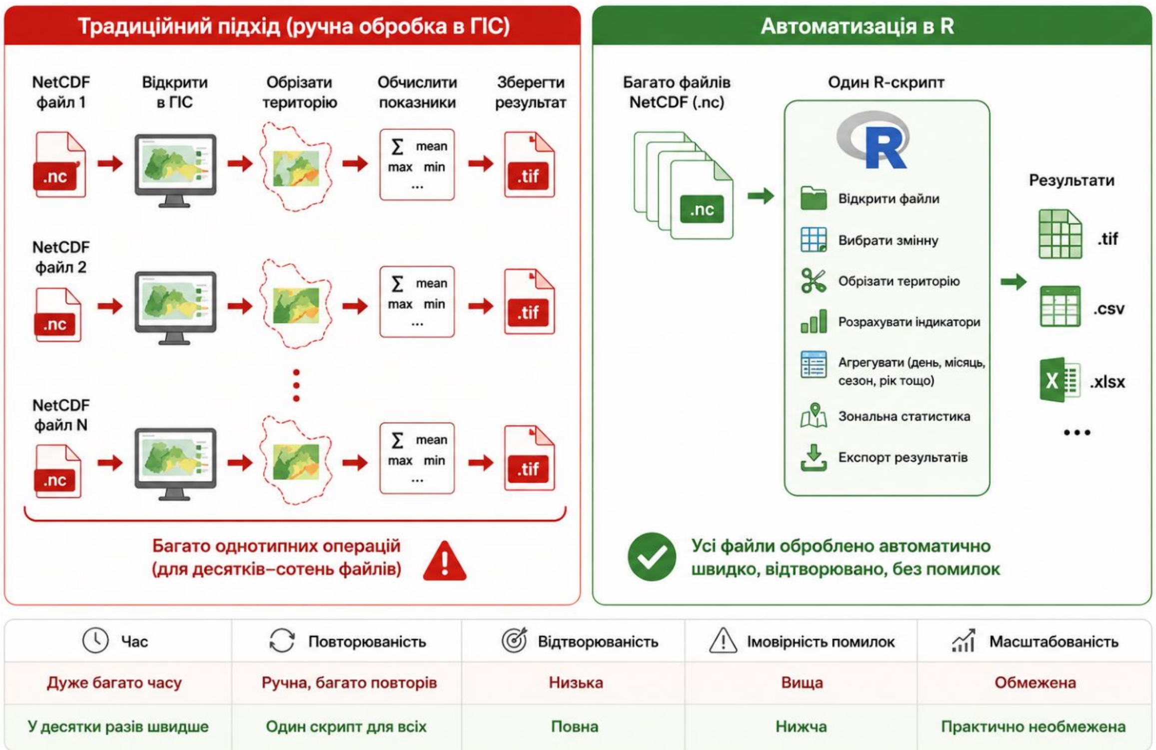

Figure 3 – Comparison of manual and automated climate data processing in GIS and R

Criterion	Manual GIS Workflow	Automation in R
Repeatability	Operations are repeated manually for each file	One script is applied to all files
Reproducibility	Depends on the documentation of manual actions	The complete workflow is recorded in code
Scalability	Limited when processing many files	Suitable for batch processing of large datasets
Visual Control	Convenient interactive review	Requires separate production of graphs or maps
Typical Application	Mapping, checking, and editing	Preprocessing, statistics, indicators, and automation

R and RStudio should therefore be used for preprocessing, automated analysis, and the calculation of climate indicators. In this workflow, GIS is used mainly at the final stage to verify, map, and visualise the results. This approach substantially reduces the time required to work with climate data, avoids repetitive manual operations, and ensures full reproducibility throughout the analysis.

10. Practical Recommendations and Good Practice

- begin by checking the metadata, variable names, units of measurement, calendar, and coordinate reference system;
- do not load more data into memory than are required for a specific calculation;
- limit the spatial and temporal extent at an early stage of processing;
- divide the script into logical stages and add clear comments;
- save intermediate results only when they are needed for quality control or reuse;
- apply consistent indicator definitions, baseline periods, and processing rules to all models and scenarios;

- check the results using descriptive statistics, graphs, and selective map review in GIS;
- record the versions of R and its packages, the source datasets, and the date on which they were downloaded;
- store scripts together with a README file or a methodological description of the workflow;
- do not interpret model projections as exact forecasts for a specific location or date.

Reproducibility

Each result should ideally be accompanied by the script, a list of input files, a description of the variables, the spatial extent, the time period, the aggregation method, the indicator definition, and the export parameters.