ESM240 – Climate Change Biology: From Models to Applications

From Models to Applications

Isaac Brito-Morales

2025-10-09

Table of contents

Preface

This eBook was written and compiled by **Isaac Brito-Morales** (*ibrito@conservation.org*) as part of the *Climate Change Biology* module within the ESM240 course.

The goal is simple: to help students and early-career scientists navigate the world of climate models — not to turn them into modelers, but to make sure they know where to find information, how to use it responsibly, and how to interpret it within an ecological and conservation context.

All code and examples are provided for educational purposes.
Please don't distribute or reuse them without permission.

> **Disclaimer**
>
> No guarantees that every line of code will run perfectly — as one of my mentors used to say, *"that's what learning (and debugging) is for."*

---

1 Welcome!

Welcome to *ESM240 – Climate Change Biology: From Models to Applications*!

This eBook was developed as part of the *Climate Change Biology* module to guide you through one of the most practical and applied aspects of climate science — learning how to understand, find, and use climate model data in real-world ecological and conservation contexts.

The purpose here isn't to make you a climate modeler. Instead, it's to help you: - Understand what climate models are and who produces them - Recognize the major CMIP versions and the meaning behind climate scenarios (RCPs and SSPs) - Know where these data live — particularly in the **Earth System Grid Federation (ESGF)** and **Metagrid** - Confidently access and visualize basic outputs, such as sea-surface temperature fields, using R

By the end of this module, you should be able to navigate the world of climate projections with confidence — knowing where to look, how to interpret model outputs, and how they connect to biological and conservation questions ( ).

This is an open educational resource built for master's students and early-career scientists who want to bridge the gap between climate modeling and applied marine science.

So, grab your coffee, open RStudio, and let's dive into the climate of the future.

1.1 Open Science: NCEAS

With this material, we are following the NCEAS data approach to make research more transparent and reproducible, which enhances the credibility, utility, and accuracy of the science used to solve global challenges.

Open science is the philosophy and practice of making data and methods accessible, replicable, and free to use, typically through computer programming tools and techniques. It helps researchers compile and analyze data more efficiently and identify solutions more quickly.

2 Setting up your computer

You will need to have both `R` and `RStudio` installed on your computer to complete this course. Although it's not essential to have the newest version of `RStudio`, **you'll need at least R version 4.0 or higher**. You might also need administrative permissions to install these programs on your machine.

After setting up `R` and `RStudio`, you'll install a few `R` packages to work with climate and environmental data. Finally, you'll download one small dataset later in the module to practice running basic analyses — nothing too heavy or complicated.

2.1 `R`

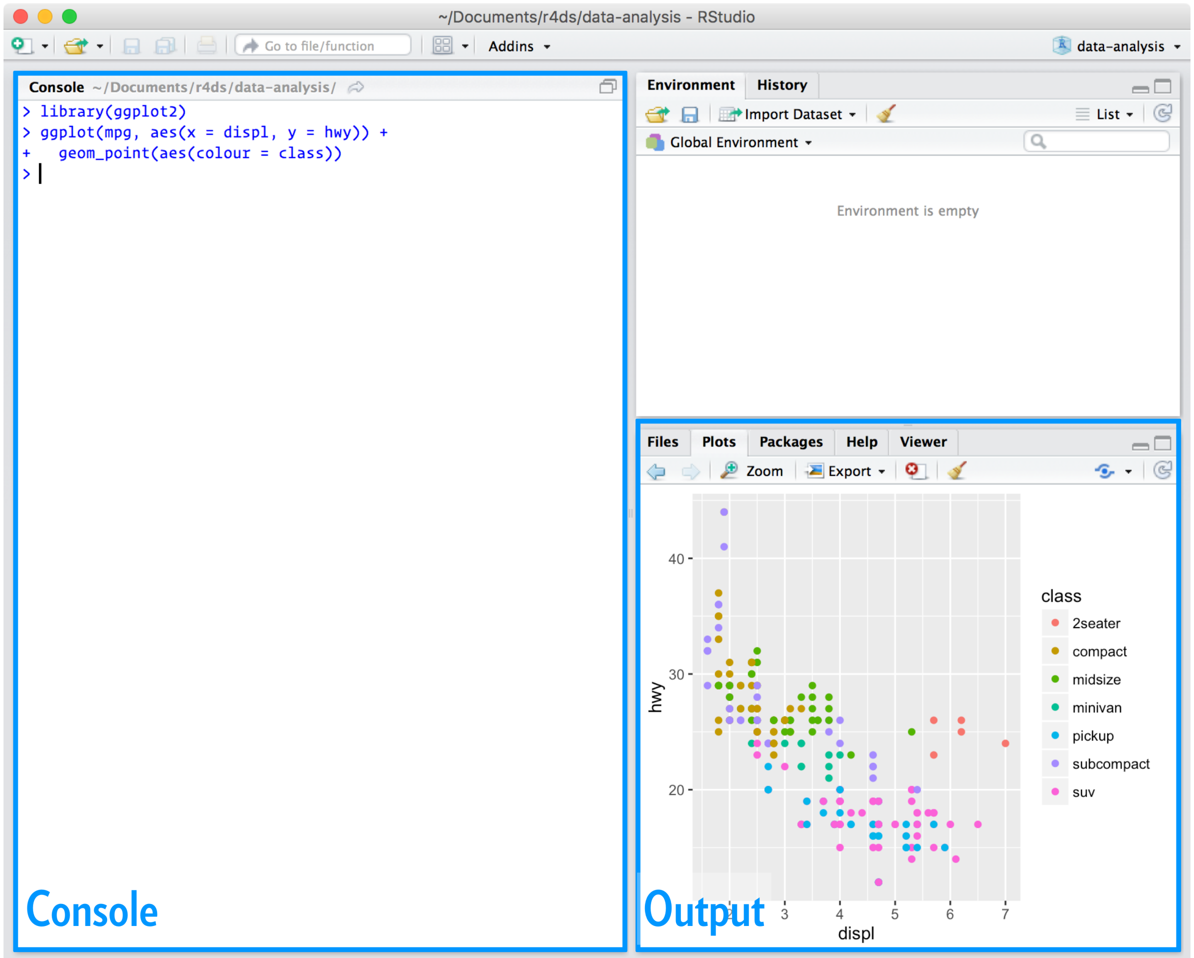
The R statistical computing environment is the core tool we'll use for data exploration and visualization in this course. You can download the latest version of `R` (version 4.2.3) from the Comprehensive R Archive Network (CRAN).

Make sure you download the correct installer for your operating system — `Windows`, `macOS`, or `Linux`. Once installed, you'll be ready to connect it with `RStudio`, the interface we'll use throughout the course.

2.2 `RStudio`

RStudio is an integrated development environment (IDE) — essentially, the workspace where you'll write and run your `R` code. It makes coding more organized and interactive, combining the console, scripts, plots, and files all in one place.

You can download the latest version of `RStudio` here. Once installed, open `RStudio` and explore its main interface — you'll find a console to run commands, a panel for your scripts, and other tabs for plots, files, and packages.



You can type R code into the *Console* and press the enter key to run code.

2.3 R packages

An R package is a collection of code, data, and documentation that extends what R can do. There are now thousands of packages on CRAN, covering everything from basic statistics to machine learning and climate modeling.

In this course, we'll only use a small handful — just enough to explore, visualize, and handle climate data efficiently. You'll see that most R workflows build on the same few "core" packages, so getting comfortable with these will serve you well beyond this class.

Navigating `R`

Because the `R` ecosystem is so extensive, the question is almost never *"can I do this in R?"* but rather *"which package should I use to do it?"*

To install the required packages, copy and paste the code below into the *Console* in `RStudio`, then press **Enter**. You'll need an internet connection, and the installation may take a few minutes.

```
# Core packages for this course
install.packages(c(
  "terra",          # raster and vector data
  "sf",             # spatial data manipulation
  "dplyr",          # data wrangling
  "ggplot2",        # plotting and visualization
  "maps",           # quick "dirty" plot
))
```

Caution

If you already have some of these installed, R will skip them automatically. No need to reinstall everything from scratch.

2.4 (Optional) Create an `R Project`

For those who want to stay extra organized, you can set up an **R Project**. An R Project connects directly to a folder on your computer, keeping your scripts, data, and outputs neatly contained in one place.

Using an R Project is a good habit for reproducible research — it keeps your workflow clean and avoids the classic *"where did I save that file?"* problem. Everything you do — data cleaning, analysis, and plotting — stays inside a single self-contained folder.

Example: Setting up an `R Project`

1. In the **File** menu, choose **New Project**
2. Select **New Directory**
3. Click **New Project**

4. Under **Directory name**, type: `esm_240_{USERNAME}`
5. Leave **Create Project as subdirectory of:** set to ~ (your home folder)
6. Click **Create Project**

Tip

If you already have your own folder system that works for you — that's fine too.
This step is optional, but it can make life easier, especially when you start juggling multiple datasets or assignments.

2.4.1 Paths 101: How R Knows Where to Look

If you're new to coding or file systems, this is worth a quick read — it'll save you a lot of confusion later on!

Every file on your computer has a **path**, which tells R where to find it.

There are two main types:

- **Absolute path:** starts from the very top (root) of your computer's file system.
  Example: `/Users/ibrito/Desktop/esm_240/`
- **Relative path:** starts from your current working directory — basically, the folder R is "looking at" right now.
  Example: `data/my_file.csv` (relative to where your project lives)

Important

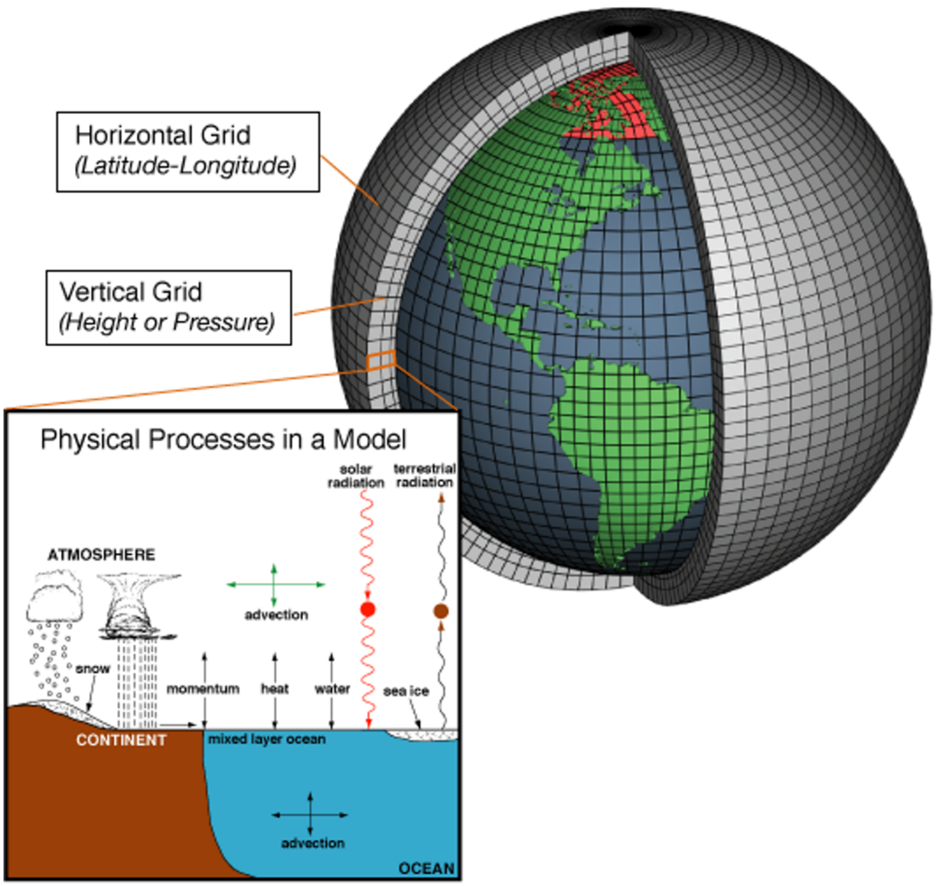
When you work inside an **R Project**, RStudio automatically sets your working directory to that project's folder.
That means your relative paths will *just work* — no need to manually tell R where to look.

Ditch the `setwd()`

Once you start using R Projects, you should almost never need to use `setwd()`.
If you find yourself typing it, pause and ask *why* — chances are, there's a cleaner way to organize your files.
(Think of it as the "Ctrl + Z" of reproducibility — best avoided if you can.)

3 Understanding `Climate Models`: The `CMIP` Framework

3.1 What is a climate model? (in one picture)



At its core, a climate model divides the Earth into a 3-D grid (latitude–longitude horizontally, levels vertically) and steps forward in time. In each grid cell the model tracks key state variables (e.g., temperature, humidity, winds, ocean currents, salinity, sea ice, carbon), and exchanges fluxes (heat, water, carbon) between neighboring cells and across components (atmosphere ocean land ice).

- Resolution: Typical global models use ~1° (~100 km) horizontal grids with tens of vertical levels
- Components: Many models are full Earth System Models (`ESMs`) that include biogeochemistry (e.g., carbon cycle), not just physics
- Outputs: Results are multi-dimensional arrays saved as `NetCDF` files (variable × lon × lat × depth × time)

Why many models? Different groups make different (reasonable) choices in numerics and physics, so we compare them within the CMIP framework (Schoeman et al. 2023).

> **Note**
>
> **Projection**, not *prediction*: Models run "what-if" scenarios (e.g., different emissions pathways). They tell us about plausible futures, not a single forecast for an exact day.

3.2 What the hell are CMIP6?

CMIP6 stands for *Coupled Model Intercomparison Project – Phase 6.*
These are large-scale simulations of the Earth system that bring together dozens of research centers around the world. Each model represents the physical, chemical, and, in some cases, biological processes of our planet's atmosphere, oceans, land, and cryosphere — the "big four" of climate (Schoeman et al. 2023)

The **IPCC Fifth Assessment Report (AR5, 2013)** was based on CMIP5, while the **Sixth Assessment Report (AR6, 2021)** uses CMIP6, which offers improved spatial resolution, new biogeochemical modules, and more realistic feedbacks among Earth-system components.

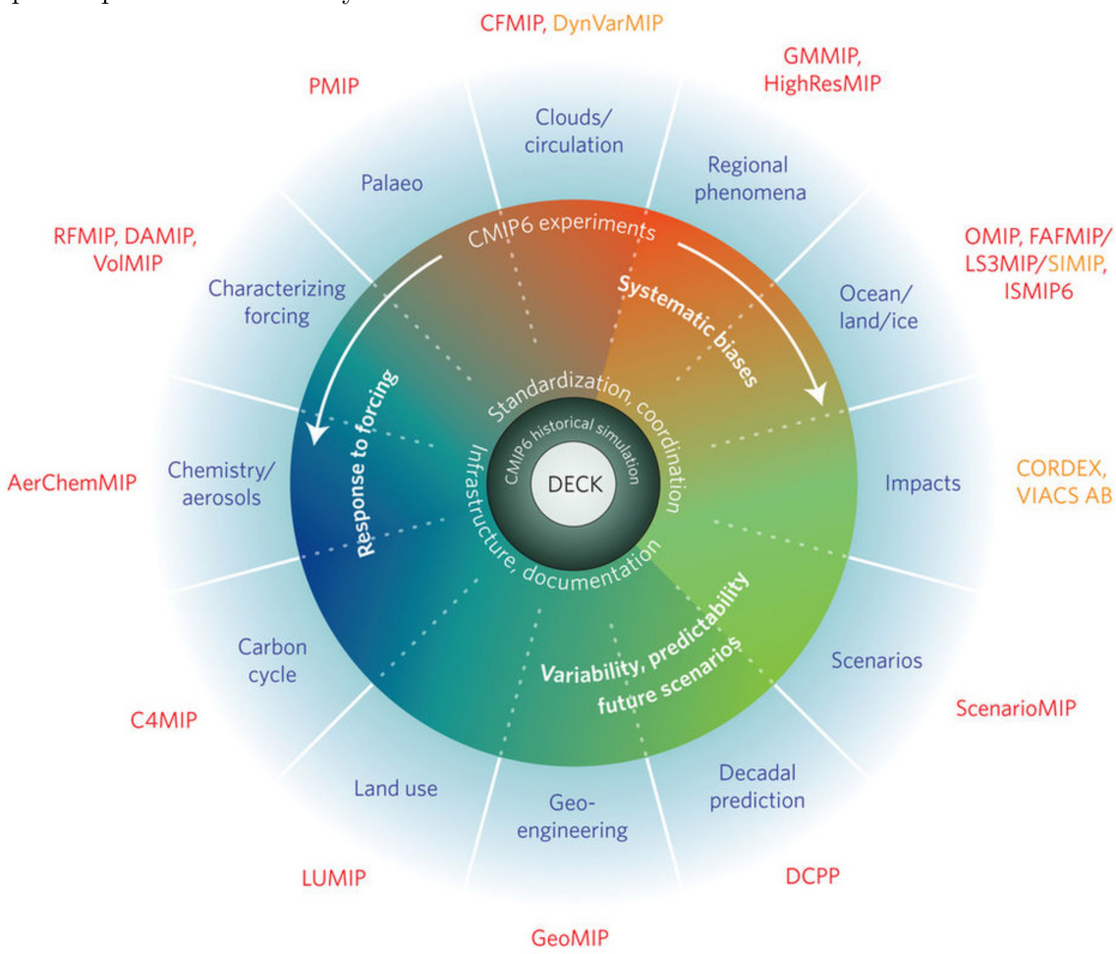
The main goal of CMIP is to explore plausible futures of the climate system under different greenhouse-gas and aerosol trajectories. Each model run represents a "what-if" experiment — what if emissions rise, stabilize, or decline?

> **Why CMIP matters (and why we spent a year writing about it)**
>
> CMIP models underpin nearly every modern climate projection — from future sea-surface warming to biodiversity redistribution.
> If you've ever used temperature or oxygen projections in ecology, you've indirectly used CMIP data.
> (And yes — this section owes more than a little to that friendly review *"Demystifying global climate models for use in the life sciences"* – which some of us may have helped write)

The CMIP framework is not a single model, but a **community experiment** that standardizes how models are run and shared, ensuring that scientists can compare results across modeling centers. These datasets are freely distributed through the Earth System Grid Federation (ESGF), which makes them available for research and teaching.

ℹ The CMIP6 "MIPs Wheel"

CMIP6 isn't a single climate model — it's a huge international collaboration that brings together more than 100 models from research centers around the world. These models are grouped into **Model Intercomparison Projects (MIPs)**, each focused on a big scientific question — for example:
How do aerosols affect clouds? How does the ocean store heat and carbon? What happens under different future scenarios?
At the center of the diagram below are a few **core experiments** that all models run to make results comparable. The outer circles show the different MIPs, each exploring a specific part of the Earth system.

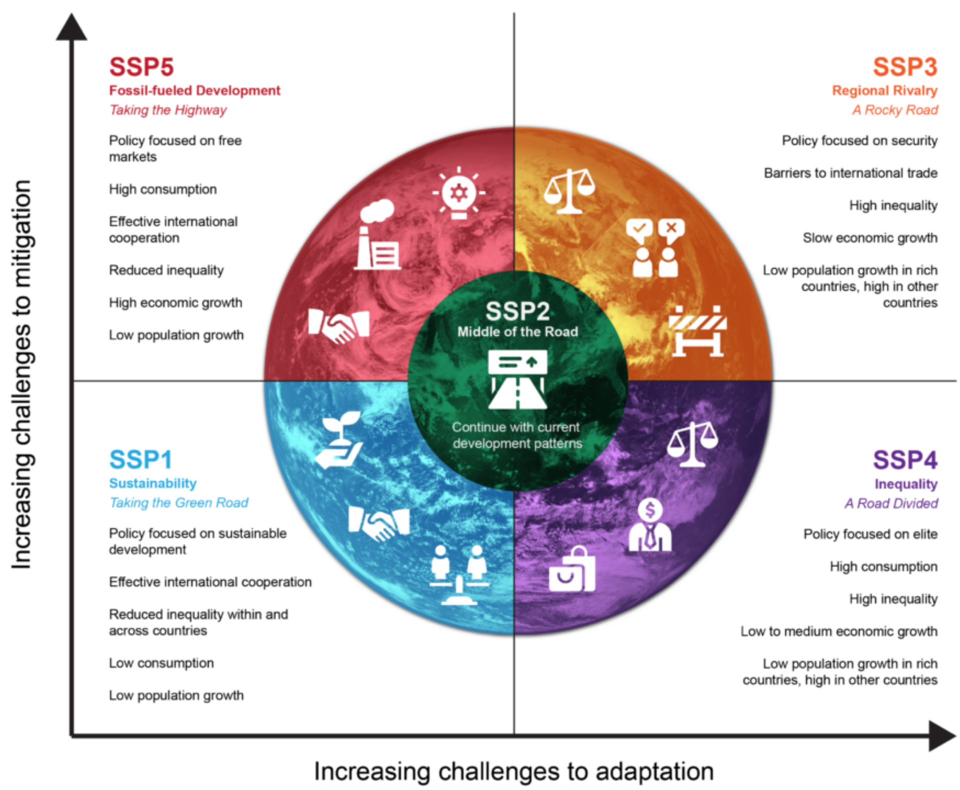


CMIP6 is what makes global climate projections possible — it provides a common framework so that results can be compared, validated, and used confidently in applied fields like ecology and conservation.
As we explain in (Schoeman et al. 2023), the system might look complicated, but it's what lets us link global climate science to questions about where and how species — and ecosystems — will respond to change.

3.3 CMIP6 Experimental Design: Climate Scenarios

Climate models explore *possible futures* — not predictions, but **scenarios** that describe how society might develop and how that would influence greenhouse gas emissions.

In CMIP5, these were called **RCPs** (Representative Concentration Pathways)
In CMIP6, they evolved into **SSPs** — *Shared Socioeconomic Pathways* — which combine both human and physical dimensions of climate change



Each SSP represents a different world:

- SSP1 – **Sustainability**: a “green road” with strong climate action
- SSP2 – **Middle of the Road**: current trends continue
- SSP3 – **Regional Rivalry**: fragmented world, high emissions
- SSP4 – **Inequality**: growing divide between rich and poor regions
- SSP5 – **Fossil-fueled Development**: rapid economic growth, very high emissions

When these socioeconomic pathways are combined with different levels of radiative forcing (how much energy the atmosphere traps), we can simulate a wide range of future climates.

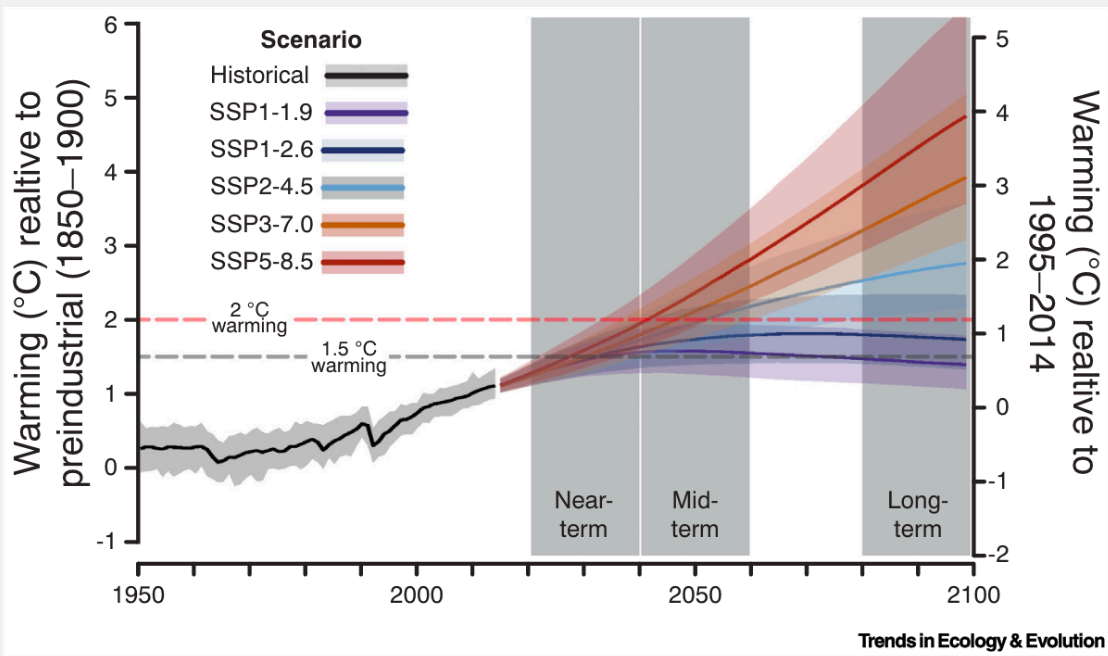


Figure I. Future temperatures under different scenarios. Global temperature change relative to preindustrial under different climate scenarios and periods considered in CMIP6, with 90% confidence intervals (as assessed in the IPCC Sixth Assessment Report [79]). Data from [80]. Abbreviations: CMIP, Coupled Model Intercomparison Project; IPCC, Intergovernmental Panel on Climate Change; SSP, socioeconomic pathway.

As shown above, temperature projections under the SSPs diverge sharply by mid-century.

For example, SSP1-2.6 keeps warming below 2 °C, while SSP5-8.5 exceeds 4 °C by 2100 — a reminder that the choices societies make now shape the climate future we’ll experience.

3.4 Downloading CMIP6 Model Data

In this section, you'll learn how to find and download real climate model outputs from **CMIP6** — the same datasets used by the IPCC in its latest assessment reports.
These files will form the foundation for your assignment, where you'll visualize and analyze global sea surface temperature (*tos*) under a future climate scenario.

3.4.1 1. Create an ESGF Account

CMIP6 data are freely available through the **Earth System Grid Federation (ESGF)** — a global network of data centers that archive Earth System Model outputs.
To download data, create an account following the official guide: https://esgf.github.io/esgf-user-support/user_guide.html.
(You can browse without logging in, but you'll need an account to download)

3.4.2 2. Navigating the Metagrid Interface

Open the **Metagrid UI**: https://aims2.llnl.gov/.
This is a modern, user-friendly front end to ESGF that lets you browse CMIP6 data using filters.

Table 1. Components of CMIP6 Earth system models archived by the Earth System Grid Federation (ESGF)[a]

Component name	Description	Details (examples in single quotes)
Activity ID	Describes the MIP to which the model output belongs. The 21 endorsed MIPs are managed independently [23] but follow the same technical requirements for all models.	MIPs most pertinent to the life sciences are 'CMIP' (historical and diagnostic simulations) and 'ScenarioMIP' (projections under different IPCC scenarios described in Box 2 in the main text). There are many others, including 'GeoMIP' (geoengineering scenarios) and 'HighResMIP' (high-resolution model outputs for assessing fine-scale phenomena such as cyclones).
Source ID	The name of the ESM.	CMIP6 currently hosts 122 models from >50 modelling centres. There are 72 ESMs in CMIP, 56 ESMs in ScenarioMIP, 8 ESMs in GeoMIP, and 40 models in HighResMIP. There are 9 ESMs that each currently provide >25 000 datasets on ScenarioMIP: (in order of decreasing number of datasets archived) 'MPI-ESM1-2-LR', 'CanESM5', 'ACCESS-ESM1-5', 'EC-Earth3', 'IPSL-CM6A-LR', 'UKESM1-0-LL', 'MIROC-ES2L', 'GISS-E2-1-G', and 'MIROC6'.
Experiment ID	The climate scenario under which the model simulations are run (see Box 2 in the main text). In CMIP6, future scenarios are denoted as SSP-RCP. The SSP prefix describes assumptions associated with socioeconomic development [22]. The RCP suffix describes greenhouse gas emissions pathways that result in specified levels of radiative forcing ($W.m^{-2}$) in 2100. All future scenarios cover 2015–2100, but in some cases extend longer (to 2300 or beyond).	Under CMIP, 'historical' simulations cover the period 1850–2014. These simulations are free-running, which means that natural variability (e.g., the El Niño/La Niña Southern Oscillation) will be out of phase with observed variability. The five most commonly used future scenarios under ScenarioMIP are [20,78]: SSP1-1.9, SSP1-2.6, SSP2-4.5, SSP3-7.0, and 'SSP5-8.5' (see Box 2 in the main text).
Model variant label	Realisation (r)	Identifies the model run, also known as the ensemble member. Different realisations ('r' numbers) denote small (and random) differences in the initial conditions for the run.
	Initialisation method (i)	Many models use a range of initialisation schemes. For example, in one scheme, different ensemble members ('r' numbers) might involve initialising from different years from the 'Control' experiment. In another, different ensemble members might be initialised from the same point of the 'Control' experiment but with a small random change added to one or more variables.
	Physics (p)	Different 'p' values indicate that some different physical processes are used in the model. For example, 'p1' and 'p2' might employ different approaches to determine when convective rainfall occurs.
	Forcing (f)	Different 'f' values indicate different forcing datasets. For example, there might be multiple different datasets available that specify anthropogenic aerosol emissions.
	Model variant numbers ('r-i-p-f' combinations) are consistent among scenarios within models, but not among models. Where specific values of any model variant descriptor are required, these details would need to be extracted from model documentation. Commonly, an ensemble from a single model across a range of variants is used to explore model sensitivity to different configurations. For example, an ensemble of realisations from a single model is useful to separate the forced signal from internal (natural) variability.	
Grid label	Each ESM is constructed around a system of grid cells that define the spatial resolution of the model on a spherical surface.	The two most common grid labels are 'gn' (outputs on the ESM natural spatial grid), and 'gr' (ESM outputs regridded onto a regular 1° grid).
Frequency	The temporal frequency at which model outputs are archived.	Not all variables are archived at all frequencies, but most are archived at monthly ('mon') intervals. Other common frequencies (in order of decreasing number of datasets archived) include annual ('yr'), daily ('day'), six-hourly ('6hr') and three-hourly ('3hr') (see also Box 3 in the main text).
Realm	The realm(s) within which the variables occur. This is important for distinguishing among variables in multiple realms.	Includes 'aerosol' (aerosol), 'atmos' (atmosphere), 'atmosChem' (atmospheric chemistry), 'land' (land surface), 'landIce' (land-based ice), 'ocean' (ocean), 'ocnBgchem' (ocean biogeochemistry) and 'seaIce' (sea ice). Despite standardised terminology, capitalisation occasionally differs among ESMs for some realms.

(continued on next page)

Metagrid UI: https://aims2.llnl.gov/ organizes CMIP6 data into facets (filters) such as model name, variable, frequency, and scenario.

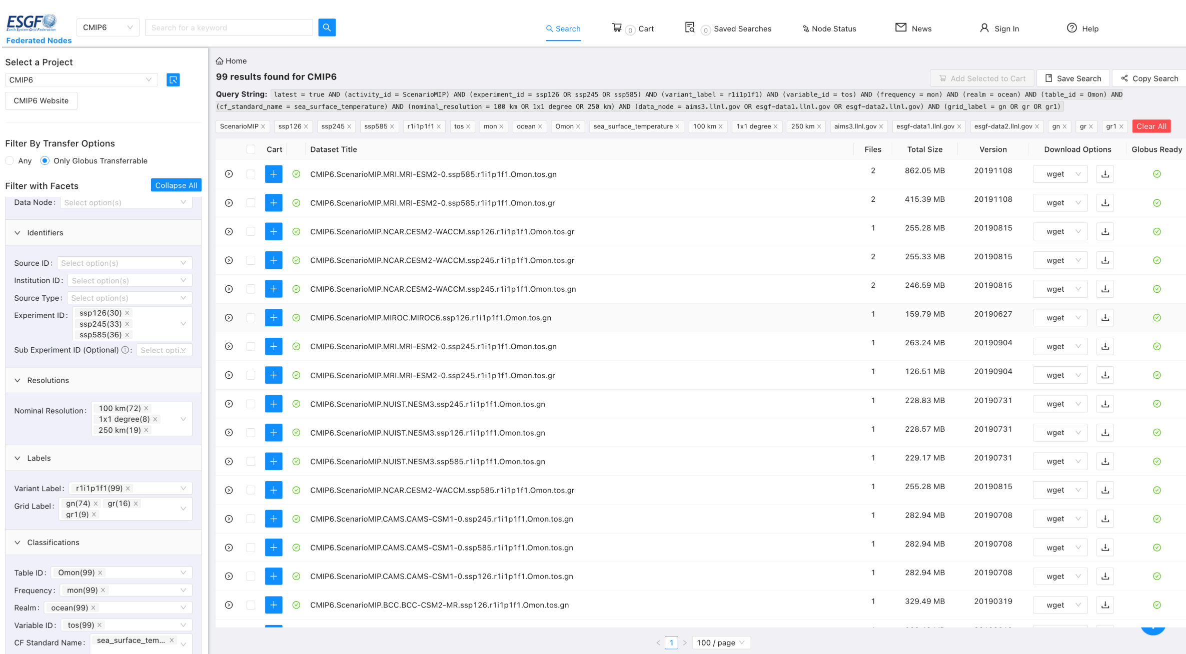
3.4.3 3. Navigating the Metagrid Interface

Component	What it means	Example
Activity ID	The experiment group (e.g., ScenarioMIP)	`ScenarioMIP`
Source ID	The model/institution	`GFDL-ESM4`
Experiment ID	The scenario (historical, ssp585, etc.)	`ssp585`
Variant Label	The ensemble member (run configuration)	`r1i1p1f1`
Frequency	Time resolution	`mon`
Realm	Earth-system component	`ocean`

3.4.4 4. Navigating the Metagrid Interface

Use the filters in the left panel. For this exercise, select:

- `Variable ID`: tos (sea surface temperature)
- `Realm`: ocean
- `Table ID`: Omon (monthly ocean outputs)
- `Frequency`: mon
- `Experiment ID`: ssp585 (future high-emission scenario)
- `Variant Label`: r1i1p1f1 (standard ensemble run)
- `Source ID`: GFDL-ESM4 (NOAA model)



You should now see a list of model outputs matching your filters. Each row corresponds to one model run for a specific scenario and variable.

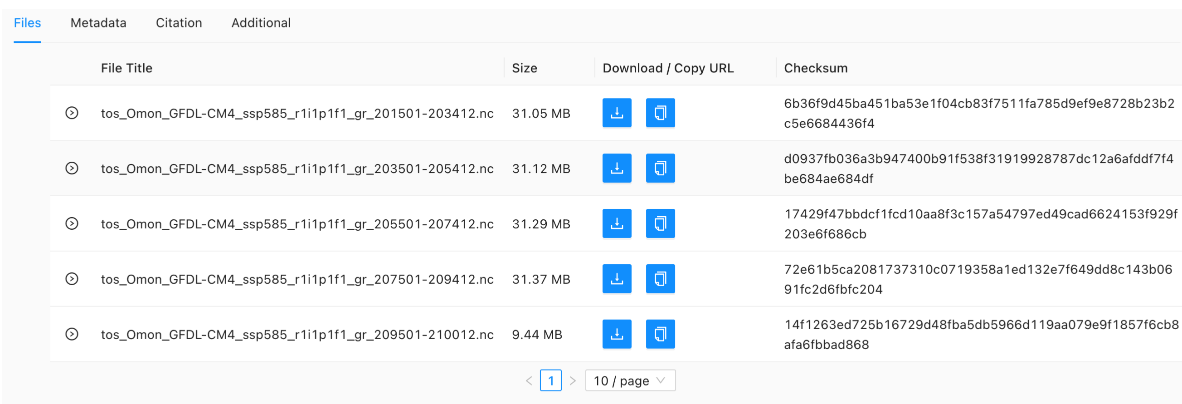
3.4.5 5. Understanding File Names

CMIP6 files follow a standard naming convention, e.g.: `tos_Omon_GFDL-ESM4_ssp585_r1i1p1f1_gr_201501-20`

Part	Meaning
`tos`	Sea surface temperature
`Omon`	Monthly ocean output
`GFDL-ESM4`	Model name
`ssp585`	Future scenario (high emissions)
`r1i1p1f1`	Ensemble member (run/physics/forcing setup)
`gr`	Grid label (regridded)
`201501-203412`	Time period covered

3.4.6 6. Downloading the File

a. Click List Files next to your chosen model entry.
b. A table will list several .nc files, each covering ~20 years.
c. Click HTTP Download for one file (e.g., 2015–2034).


Files Metadata Citation Additional

File Title	Size	Download / Copy URL	Checksum
tos_Omon_GFDL-CM4_ssp585_r1i1p1f1_gr_201501-203412.nc	31.05 MB		6b36f9d45ba451ba53e1f04cb83f7511fa785d9ef9e8728b23b2c5e6684436f4
tos_Omon_GFDL-CM4_ssp585_r1i1p1f1_gr_203501-205412.nc	31.12 MB		d0937fb036a3b947400b91f538f31919928787dc12a6afddf7f4be684ae684df
tos_Omon_GFDL-CM4_ssp585_r1i1p1f1_gr_205501-207412.nc	31.29 MB		17429f47bbdcf1fcd10aa8f3c157a54797ed49cad6624153f929f203e6f686cb
tos_Omon_GFDL-CM4_ssp585_r1i1p1f1_gr_207501-209412.nc	31.37 MB		72e61b5ca2081737310c0719358a1ed132e7f649dd8c143b0691fc2d6fbfc204
tos_Omon_GFDL-CM4_ssp585_r1i1p1f1_gr_209501-210012.nc	9.44 MB		14f1263ed725b16729d48fba5db5966d119aa079e9f1857f6cb8afa6fbbad868

1 10 / page

Each file is ~130 MB. Do not download all time periods — one file is enough for this assignment.

Assignment Hint

Use this setup: - Model: GFDL-ESM4 - Scenario: ssp585 - Variable: tos (sea surface temperature) - Period: 2015–2034 You'll use this file in the next section to open, visualize, and analyze global SST in R. *Pro tip: Start filtering by variable — it's the quickest way to narrow the search.*

Assignment Hint

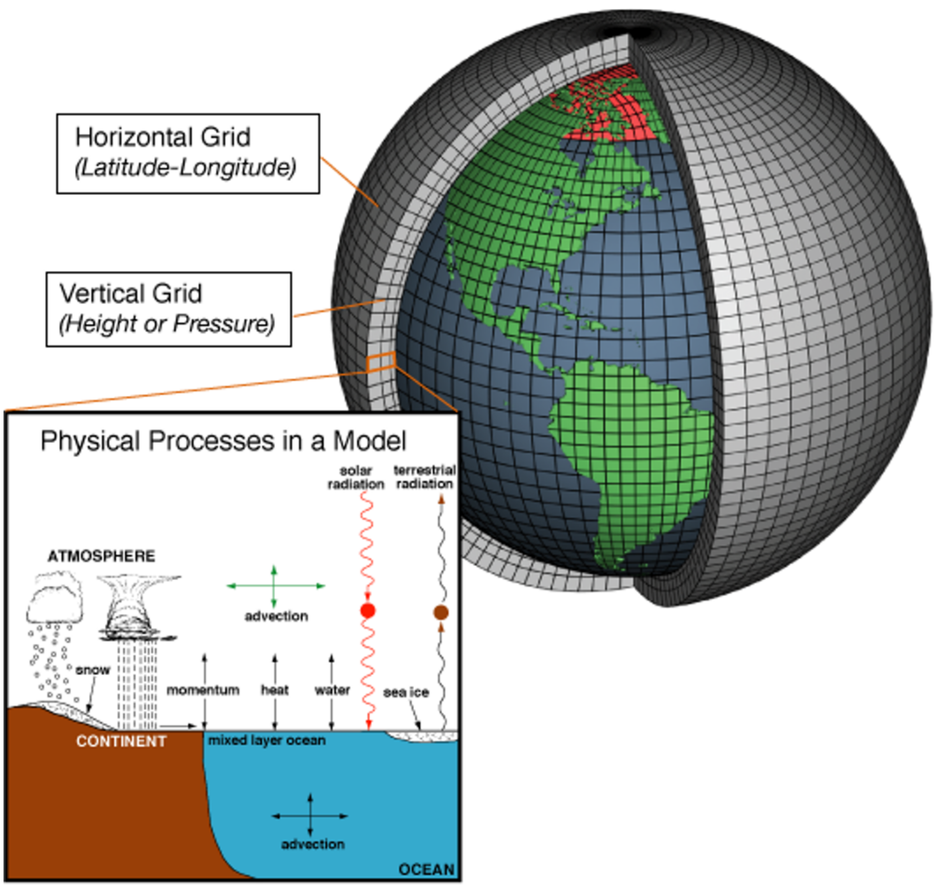
No downloads? Make sure you're logged into `ESGF`.
Empty results? Clear filters and re-apply `Variable` → `tos` first.
Slow/failed download? Try another `ESGF` node mirror listed under `"Download Options"`

4 From `NetCDF` to a Map in `R`

4.1 CMIP6 data, explained in 10 minutes

4.1.1 What is `NetCDF` (and why CMIP6 uses it)?

NetCDF is a container for scientific arrays (e.g., temperature) plus dimensions (lon, lat, depth, time) and metadata (units, CF names).
CMIP6 outputs are saved as NetCDF because it's compact and self-describing. We won't cover NetCDF internals here—think of it as the standard file you download and open.



4.1.2 What is a `raster` (in `R`)?

A **raster** is how gridded data are handled in `R` (via `{terra}`): it's what you plot, crop, reproject, and compute on.
In one line: *NetCDF is the storage format; a raster is how we work with it in R.*

4.2 Load libraries

```
library(terra)
library(maps)
```

1) Load the NetCDF file

```
# Change this to YOUR downloaded file
f <- "path/to/your/tos_Omon_GFDL-ESM4_ssp585_r1i1p1f1_gr_201501-203412.nc"

# Lazy connection to the NetCDF file (does not load everything into memory)
r <- rast(f)
```

2) Detect the variable

```
# Short variable name(s) in the NetCDF (e.g., "tos", "thetao", etc.)
vn <- tryCatch(terra::varnames(r), error = function(e) NA_character_)
# Use the first variable stack in the file (keeps this example generic)
r_var <- r[[1]]
```

3) Print key metadata (model, scenario, ensemble, units, resolution)

```
cat("\n================== METADATA ==================\n")
print(r)

# Parse common CMIP6 filename parts
fname   <- basename(f)
parts   <- unlist(strsplit(fname, "_"))

model    <- if (length(parts) >= 3) parts[3] else NA_character_
scenario <- if (length(parts) >= 4) parts[4] else NA_character_
ensemble <- if (length(parts) >= 5) parts[5] else NA_character_
```

```r
# Units may not always be present-handle gracefully
u <- tryCatch(units(r_var)[1], error = function(e) "unknown")

cat("\nModel name:", model,
    "\nEnsemble member:", ensemble,
    "\nScenario:", scenario,
    "\nNative resolution (lon x lat, degrees):", paste(terra::res(r_var), collapse = " x "),
    "\nVariable name(s):", if (is.null(vn)) "unknown" else paste(vn, collapse = ", "),
    "\nVariable units:", u,
    "\n============================================\n\n")
```

4) Plot: first time slice (quick look)

```r
# If a time dimension exists, take the first layer
if (nlyr(r_var) > 1) r_var <- r_var[[1]]

# Rotate longitudes to [-180, 180] for a clean world map
r_plot <- terra::rotate(r_var)

# Try to extract a timestamp (may be NULL depending on file)
tstamp <- tryCatch(as.character(terra::time(r_plot)[1]), error = function(e) "")

# Save a quick PNG to your project folder
png("quick_var.png", width = 1000, height = 600, res = 120)
plot(r_plot, main = paste("Quick Plot -", ifelse(is.null(vn), "variable", vn[1]), tstamp))
maps::map("world", add = TRUE)
dev.off()
```

5) Compute a global mean (first time slice)

```r
mean_val <- terra::global(r_plot, "mean", na.rm = TRUE)[[1]]

cat("Variable plotted:", ifelse(is.null(vn), "unknown", vn[1]), "\n")
cat("Global mean value (first time step):", round(mean_val, 3), u, "\n")
cat("PNG saved:", normalizePath("quick_var.png"), "\n")
```

References

Schoeman, David S., Alex Sen Gupta, Cheryl S. Harrison, Jason D. Everett, Isaac Brito-Morales, Lee Hannah, Laurent Bopp, Patrick R. Roehrdanz, and Anthony J. Richardson. 2023. "Demystifying Global Climate Models for Use in the Life Sciences." *Trends in Ecology & Evolution*, May. https://doi.org/10.1016/j.tree.2023.04.005.