



climate change initiative

tertiary education resources

CLIMATE ANALYSIS TOOLBOX, Cate

Tutorial

Contents

CLIMATE ANALYSIS TOOLBOX, Cate	4
Fast facts.....	4
Brief description	4
Intended learning outcomes	5
Summary of activities.....	6
Health and safety.....	7
CHAPTER ONE.....	8
CHAPTER TWO	11
Background	11
Scope.....	11
CHAPTER THREE.....	13
System Requirements	13
Installation	13
Cate SaaS.....	13
Cate Desktop.....	14
CHAPTER FOUR.....	15
Panels	15
Views	17
Menu Bar.....	17
CHAPTER FIVE.....	18
Data Sources.....	18
ESA CCI Open Data Portal.....	18
Local Data Sources.....	20
CHAPTER SIX.....	23
Open and visualise data.....	23
Processing data.....	28
CHAPTER SEVEN	34
What are Jupyter Notebooks?.....	34
Installing Jupyter.....	37
Step 1 Install Jupyter using Anaconda.....	37
Step 2 Install Cate and its requirements.....	38
Step 3 Start a Jupyter Notebook	38
Step 4 Create and try out try out your first notebook	38
References.....	39
Notes.....	39

climate change initiative CLIMATE ANALYSIS TOOLBOX, Cate
<https://climate.esa.int/educate/>

Activity concepts developed by University of Twente (NL)

The ESA Climate Office welcomes feedback and comments
<https://climate.esa.int/helpdesk>

Produced by the ESA Climate office
Copyright © European Space Agency 2021

CLIMATE ANALYSIS TOOLBOX, Cate

Fast facts

Subjects: tutorial, to provide work experience with the Climate analysis Toolbox (Cate)

Type: tutorial

Complexity: medium

Lesson time required: 8 hours

Cost: none

Location: indoors

Includes the use of: internet, and the CCI Toolbox (Cate)

Keywords: climate change, essential climate variables, satellite

Brief description

This tutorial demonstrates how to use Cate Desktop interface to handle and analyse climate records of essential climate variables produced by the Climate Change Initiative programme of the European Space Agency

The exercise presented in this tutorial is based on Cate desktop version 2.0.0

Intended learning outcomes

After following the steps described in this tutorial, students will be able to:

- build practical work experience with the Climate Analysis Toolbox (Cate)
- retrieve and pre-process data from the climate records of essential climate variables
- analyse data from the climate records of essential climate variable

Summary of activities

	Title	Description	Outcome	Requirements	Time
1	Climate analysis Toolbox (Cate)	This tutorial demonstrates how to use Cate Desktop interface to handle and analyse climate records of essential climate variables produced by the Climate Change Initiative programme of the European Space Agency	<p>Build practical work experience with the Climate Analysis Toolbox (Cate)</p> <p>Retrieve and pre-process data from the climate records of essential climate variables</p> <p>Analyse data from the climate records of essential climate variables.</p>	<p>Desktop or laptop with ~25GB of free storage on the hard disk</p> <p>Good to excellent internet connection</p>	8 hours

Times given are for the main tutorial's activities. They include time for exploring the Data Open Portal, but not experimenting with the Climate from Space web application.

Health and safety

In all activities, we have assumed you will continue to follow your usual procedures relating to the use of common equipment (including electrical devices such as computers and readers), movement within the learning environment, trips and spills, first aid, and so on. Since the need for these is universal but the details of their implementation vary considerably, we have not itemised them every time. Instead, we have highlighted hazards particular to a given practical activity to inform your risk assessment.

All the activities involve the use of Climate analysis Toolbox (Cate) and online resources from the Open Data Portal. Cate connects to Open Data Portal and accesses huge data. Be cautious when using the option of Caching Data Resources as this may take up a considerable amount of space on your desktop. We provide sample data set to work with, which also requires some space on your hard drive. If you are not able – or do not wish – to use your desktop storage, then you can work with Cate Software -as-a-Service on the cloud. In this case you are reminded of your local Internet and data access safety rules.

CHAPTER ONE

Climate from Space

ESA satellites play an important role in monitoring climate change. The *Climate from Space* (CfS) web app (cfs.climate.esa.int) is an online resource that uses illustrated stories to summarise some of the ways in which our planet is changing, and highlight the work of ESA scientists.

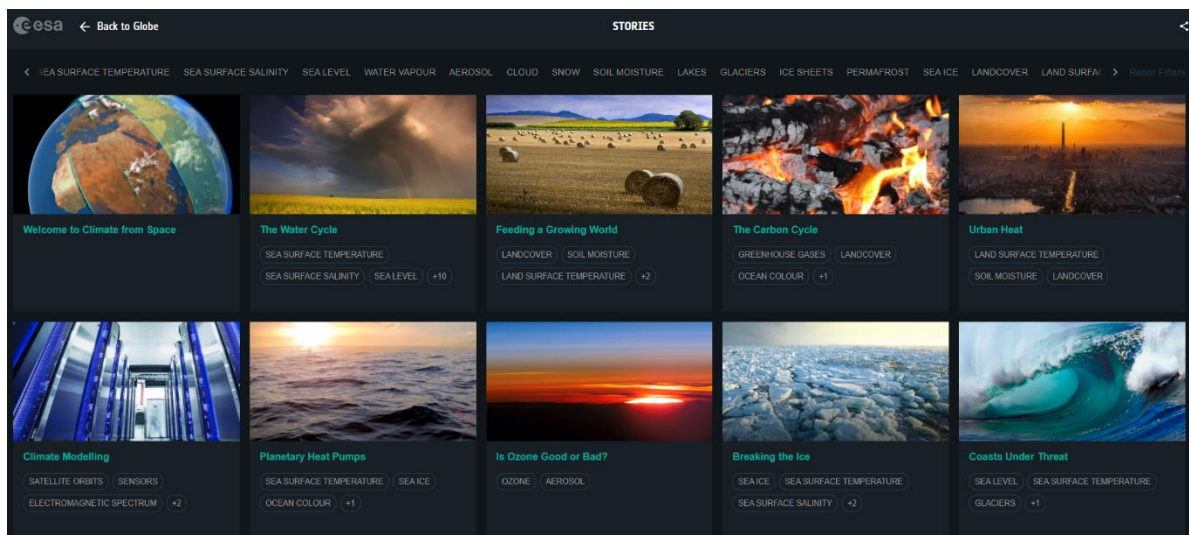


Figure 1: Stories in Climate from Space (ESA CCI)

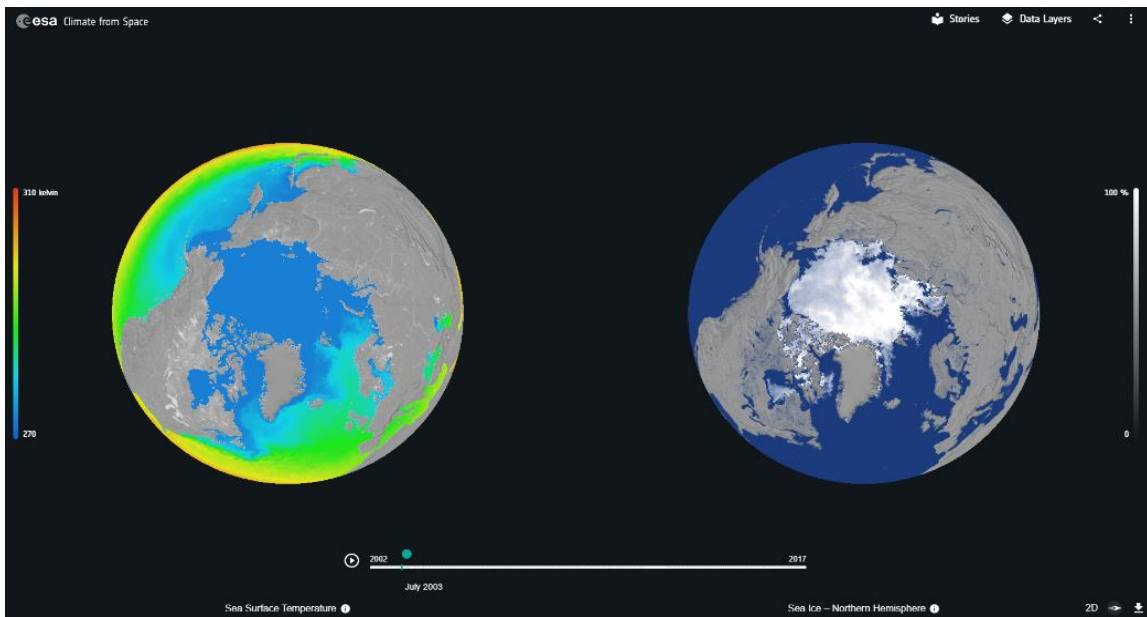


Figure 2: Exploring sea surface temperatures and sea-ice extent in the Climate from Space data viewer (ESA CCI)

ESA's Climate Change Initiative programme produces reliable global records of some key aspects of the climate known as essential climate variables (ECVs). The Climate from Space data viewer allows you to find out more about the impacts of climate change by exploring this data for yourself.

The Climate Analysis Toolbox for ESA (Cate), on the other hand, is a cloud-enabled computing environment to analyse, process and visualise climate data.

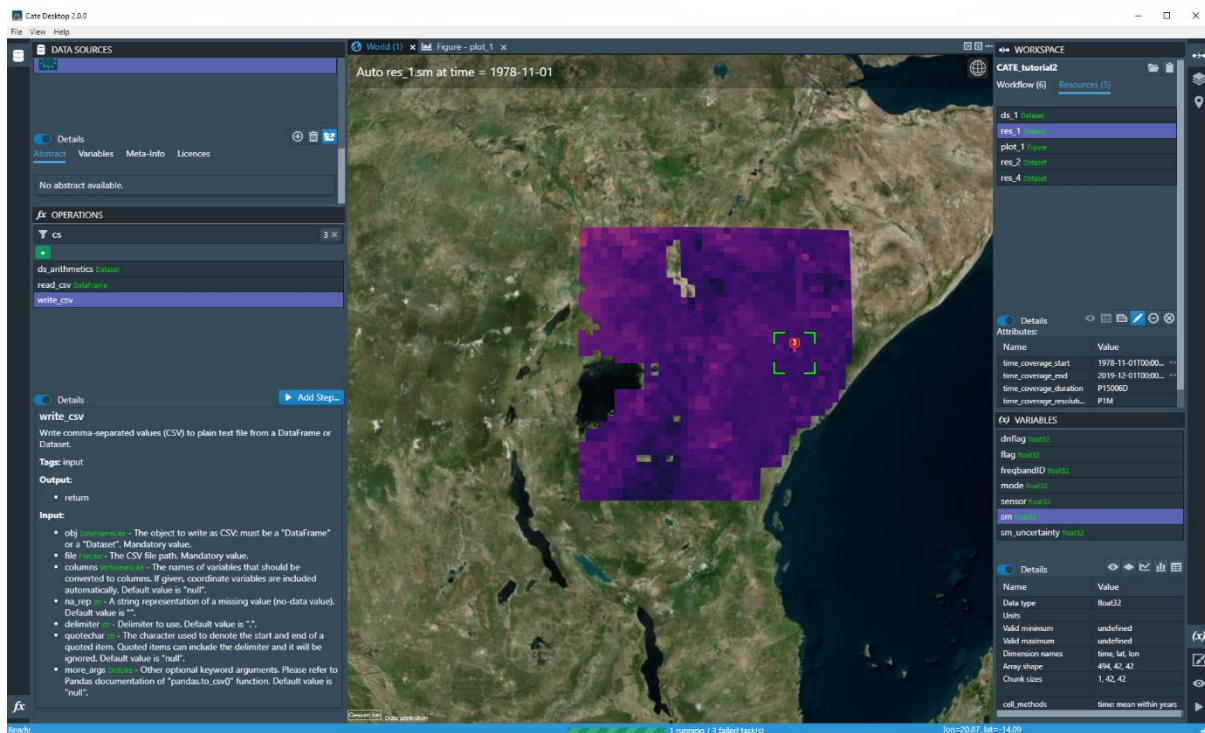


Figure 3: The CCI Toolbox (Cate).

CHAPTER TWO

What is CATE?

Background

The Climate Change Initiative (CCI) programme is the response from the European Space Agency (ESA) to the need for reliable and consistent climate data as formulated by the United Nations Framework Convention on Climate Change (UNFCCC). The programme consists of twenty-three parallel projects, each generating a climate data record for a well-defined essential climate variable (ECV) using the long-term global satellite data archives that ESA has established with its Member States over the past forty years. The CCI Open Data Portal and the Climate Analysis Toolbox for ESA (Cate) are the two main technical support projects within the programme. The CCI Open Data Portal (ODP) provides a single point of harmonised access to a subset of mature and validated data on Essential Climate Variables (ECVs). Cate provides tools that support visualisation, analysis and processing of ECVs and other climate data products.

Scope

Cate comprises four major software interfaces:

1. Cate Software-as-a-Service (SaaS) provides users access to the Cate software without any installation and configuration. Cate SaaS also provides some computational resources free of charge, however, service capacities might be throttled depending on the number of concurrent users logged into the system. In the future, this will be the recommended way to use Cate for most users.
2. Cate Desktop is similar to SaaS in appearance and functionality but is a cross-platform desktop application for users who wish to use Cate predominantly with their local data sources. The application can also connect to the remote Cate services of Cate SaaS.
3. The Cate Command-Line Interface (CLI) from a local Cate installation can be used to access and process data through a command shell or console terminal. Almost all of Cate functionality except interactive visualisation is accessible through its CLI when installed locally. The CLI can be used by users who wish

to use Cate to process and analyse data, but use their own preferred software for further analysis and visualization. The CLI can also produce static plots as supported by commonly used matplotlib. This CLI is also suited for batch processing scripts for automation.

4. The Cate Python API enables the use of Cate functions in Python programmes and may also be used to extend Cate.

Although this tutorial demonstrates how to use **Cate Desktop interface**¹ version 2.0.0 to handle and analyse ECVs data records from the Climate Change Initiative (CCI) programme, it also applies to Cate Software-as-a-Service (SaaS). The data used in this tutorial were, however, downloaded and analysed on a local machine.

¹ From a user's perspective, there are no differences between the Cate SaaS and Cate Desktop interfaces

CHAPTER THREE

Prerequisites

For installing Cate with its different interfaces the reader is referred to the User Manual. In this section, we will explain how to install the Cate SaaS and Desktop.

System Requirements

Hardware: It is recommended to use an up-to-date computer, with at least 8GB of RAM and a multi-core CPU. The most important bottlenecks will first be the data transfer rate from local data caches into the executing programme, so it is advised to use fast solid-state disks. Secondly, the internet connection speed matters, because Cate will frequently have to download data from remote services to cache it locally.

Operating Systems: Cate is supposed to work on up-to-date Windows, Mac OS X, and Linux operating systems.

Installation

Cate SaaS

For the Cate SaaS interface there is no installation needed, just type on your web browser <https://cate.climate.esa.int/>. You will then be directed to the Cate service website (Figure 4Figure 4). For the SaaS interface click on the Cate Cloud Service.

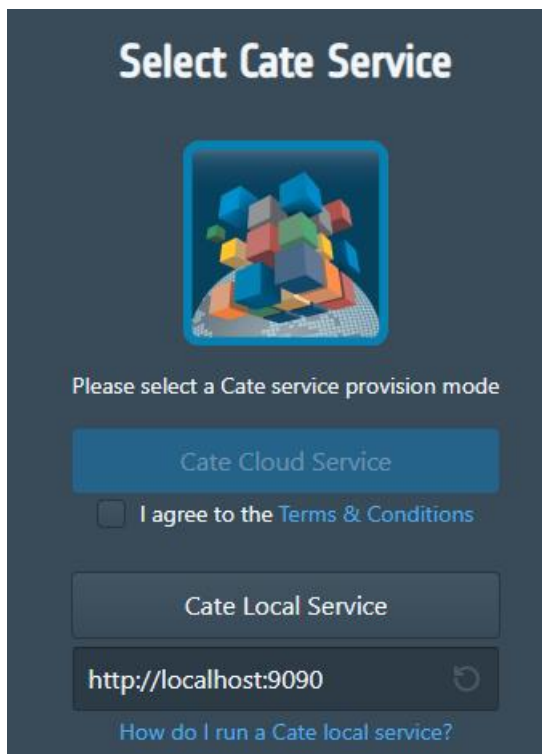


Figure 4: Cate Service website.

Cate Desktop

For Cate Desktop, we provide an installer for Windows, Mac OS X, and Linux operating systems. The Cate Desktop installer will also ensure the Cate Python package is installed. If it cannot find an existing or compatible Cate Python package, it will install a new or update an existing one.

The Cate Desktop installer for your platform is available from the Cate website. Should the website be unavailable, you can get the installer directly from the releases page in Cate Desktop's GitHub repository:

- Cate-desktop-2.x.y.dmg and for OS X;
- Cate-desktop-2.x.y-x86_64.AppImage for Linux;
- Cate-desktop-setup-2.x.y.exe for Windows.

All Cate Desktop installers are quite light-weight and executed by double-clicking them. They don't require any extra user input up to the point where no existing or compatible Cate Python package is found.

In this case, Cate's setup process will start, follow the steps in the installation wizard, for more details the user is advised to consult the User Manual.

CHAPTER FOUR

The Graphical User Interface of Cate

Panels

When run for the first time, the initial layout and position of the panels are shown in Figure 5, which contains data sources, operations, resources/datasets, and variables:

- On the upper left, the DATA SOURCES panel to browse, download and open both local and remote data sources, including data from ESA CCI Open Data Portal;
- On the lower left, the OPERATIONS panel to browse and apply available operations;
- On the upper right, the WORKSPACE panel to the browser and select available resources and workflow steps resulting from opening data sources and applying operations;
- On the lower right, the VARIABLES panel to browse and select the geophysical variables contained in the selected resource.

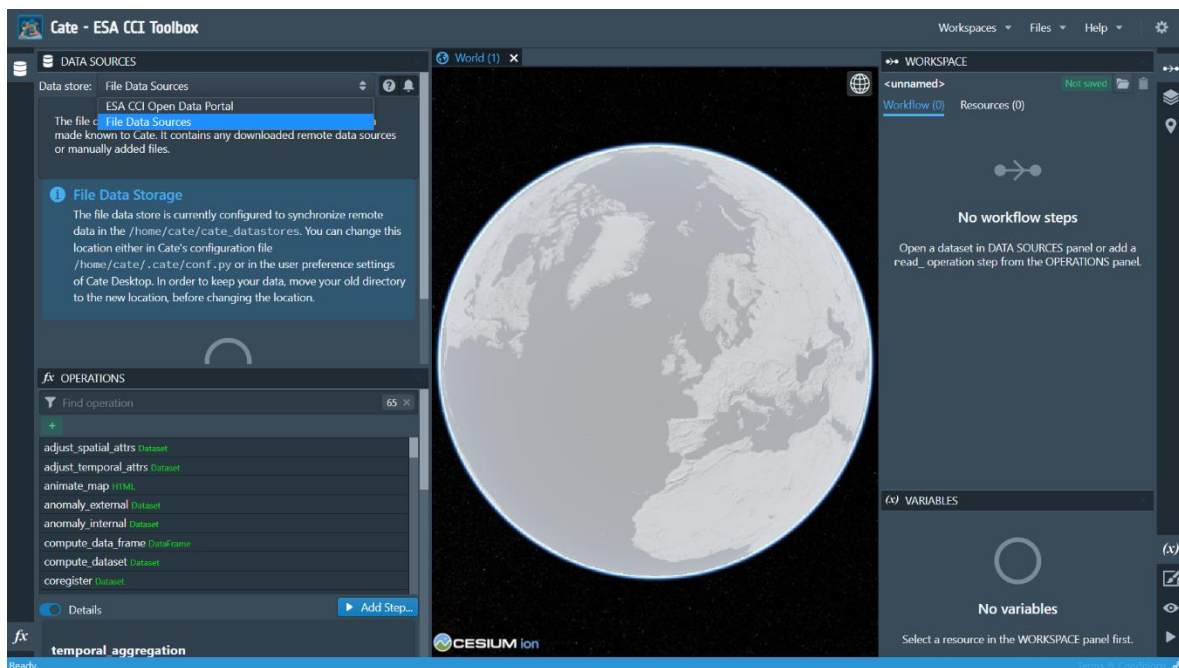


Figure 5: Graphical User Interface (GUI) of Cate.

Other panels are initially hidden. These are:

- On the upper right, the LAYERS panel, to manage the imagery layers displayed on the active World view;
- On the upper right, the PLACEMARKS panel, to manage user-defined placemarks, which may be used as input to various operations, e.g., to create time series plots;
- On the lower right, the STYLES panel, to adjust the styles of the selected layer or entity;
- On the lower right, the VIEWS panel, to display and edit properties of the currently active view. It also allows for creating new World views;
- On the lower right, the TASKS panel, to list and possibly cancel running background tasks.

Each panel's visibility can be controlled by left- and right-most panel bars. Click on a panel icon to toggle its visibility. Between two panels, there are invisible, horizontal split bars. Move the mouse pointer over the split bar to see it turning into a split cursor, then drag to change the vertical split position. In a similar way, there are invisible, vertical split bars between the tool panels and the views area. Move the mouse cursor over them to find them.

Views

The central area is occupied by views that can be arranged in rows and columns. Cate currently offers three view types:

- The world view, displaying imagery data originating from data variables and placemarks on either a 3D globe or a 2D map;
- The table view, displaying tabular resource and variable data in a table;
- The figure view, displaying plots from special figure resources resulting from the various plotting operations.

There may be multiple views stacked in a row of tabs, where each tab represents a view. One view within a tab row is selected and visible. The selected view can be split horizontally or vertically by dedicated icon buttons on the right of the tab row header. A split view can be stacked again by the drop-down menu on the right-most position of the row tab header.

There is always a single active view indicated by the blueish view header text. To activate a view, click its header text. The active view provides a context for various commands, for example, all interactions with the LAYERS and VIEW panels are associated with the active view.

Initially, a single World view is opened and active.

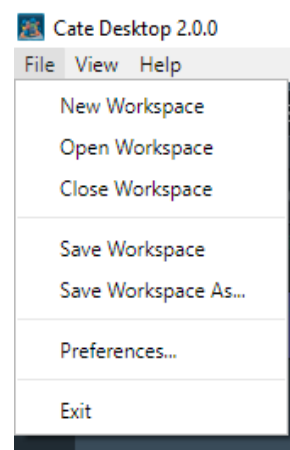


Figure 6: Cate Desktop's File menu

Menu Bar

Cate's menu currently comprises (Figure 6) the File, View, and Help menus. The File menu comprises Workspace-related commands and allows setting user Preferences:

Menu item	Description
New Workspace	Creates a new <i>scratch</i> workspace. Scratch workspaces are not-yet-saved workspaces.
Open Workspace	Opens an existing workspace. Will open a dialog to select a workspace directory.
Close Workspace	Close current workspace and create a new scratch workspace.
Save Workspace	Save current workspace in its directory. Will delegate to Save Workspace As if it hasn't been saved before.
Save Workspace As	Opens a dialog to choose a new <i>empty</i> , directory in which the current workspace data will be saved. This will become the current workspace directory.
Preferences	Opens a dialog where users can adjust various settings according to their preferences. See also Preferences Dialog .
Exit / Quit	Exits the application

CHAPTER FIVE

Data Access

Data Sources

The DATA SOURCES panel is used to browse, download and open both local and remote data sources published by data stores.

Using the drop-down list located at the top of the panel, it is possible to switch between the currently available data stores. By default, Cate uses the CCI Open Data Portal (ODP) remote data store which provides access to all published CCI datasets. There is also a local data store, which is used to synchronise remote data or to add any other data sources to Cate.

Selecting a data source entry will allow displaying its details, namely the available (geo-physical) variables and the meta-data associated with the data source.

ESA CCI Open Data Portal

In order to start working with remote data from the ESA CCI Open Data Portal data store, there are two options which are explained in the following:

- 1 Download the complete remote dataset or a subset and make it a new local data source available from the local data store. Open the dataset from the new local data source. This is currently the recommended way to access remote data as local data stores ensure sufficient I/O performance and are not bound to your internet connection and remote service availability.
- 2 Open the remote dataset without creating a local data copy. This option should only be used for small subsets of the data, e.g., time series extractions within small spatial areas, as there is currently no way to observe the data rate and status of data elements already transferred. (Internally, we use the OPeNDAP service of the ESA CCI Open Data Portal.)

Now select the ESA CCI Open Data Portal, and scroll down and select the Dataset Aerosol². Cate will provide an explanation of the data clarifying the data source and the algorithm used to produce it.

Please note that accessing the ESA CCI ODP data store from Cate provide only a subset of the data available. The full archive of data is accessible via the <https://climate.esa.int/odp/>.

Now open the data by clicking on the Open Data Source button (the red box in Figure 7).

You can select (and you are advised to do so) region, time, and variable constraints. Select a small region and the full-time record of observation. You can also provide a name for the new data source. The operation may take considerable time to download the data for the first time. Checking the box Cache data source will require a lot of free space on your local system. By default, Cate stores this data in the user's home directory. On Linux and Mac OS, that is `~/.Cate/data_stores`, on Windows it is `%USER_PROFILE%\Cate\data_stores`. Use the Preferences Dialog to set an alternative location. After confirming the dialog, a download task will be started, which can be observed in the TASKS panel. Once the download is finished, a notification will be displayed and a new local data source will be available for the local data store.

² Note there are many data sources for one variable, which has to do with the satellites used and the employed algorithms used, at this stage just select any one

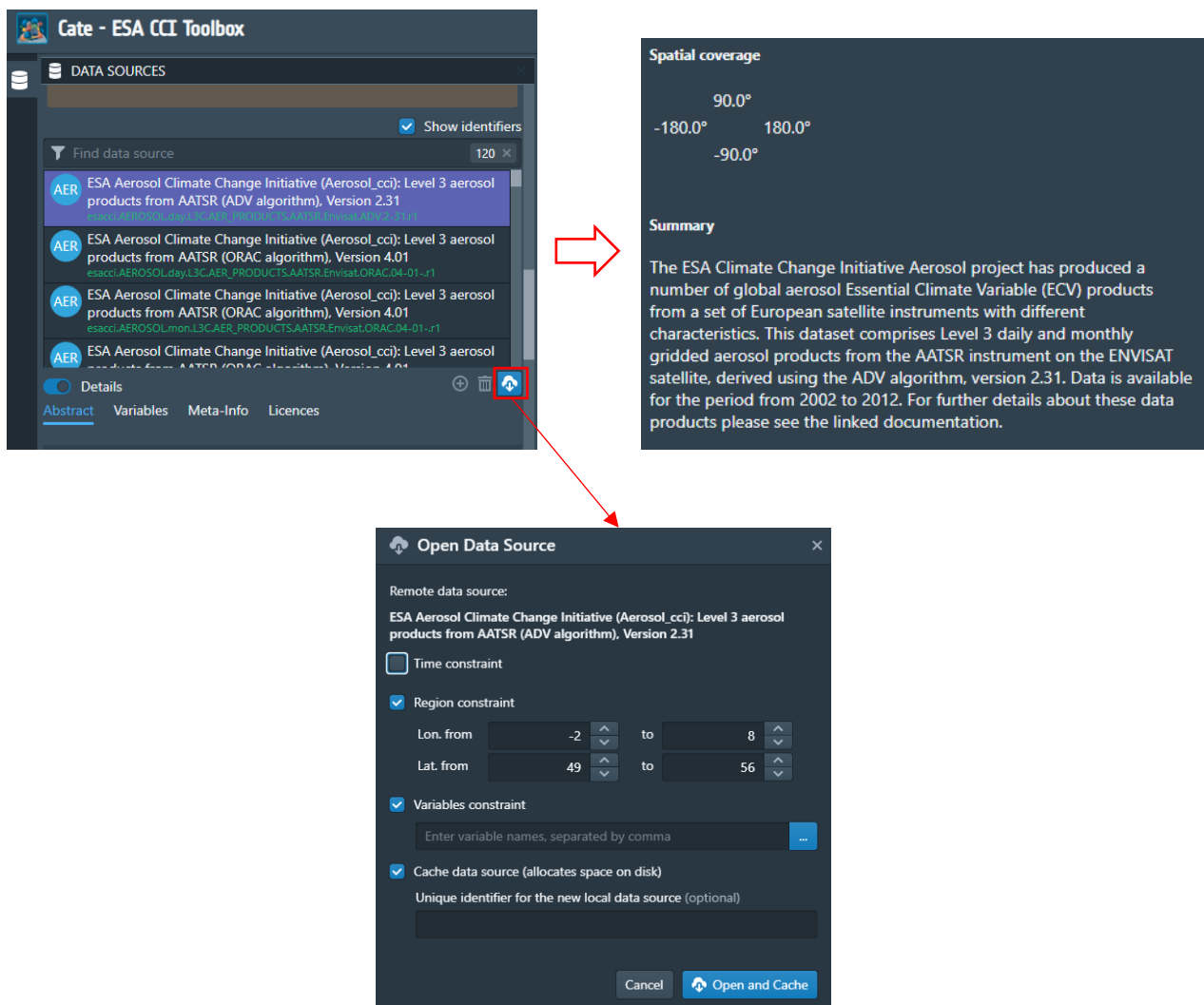


Figure 7: Adding a data source and opening it with constraints.

Local Data Sources

The full archive of data is accessible via the <https://climate.esa.int/odp/>. From this website, you can select the data you are interested in (Figure 8Figure 8), let us use the Aerosolⁱ data for this example.

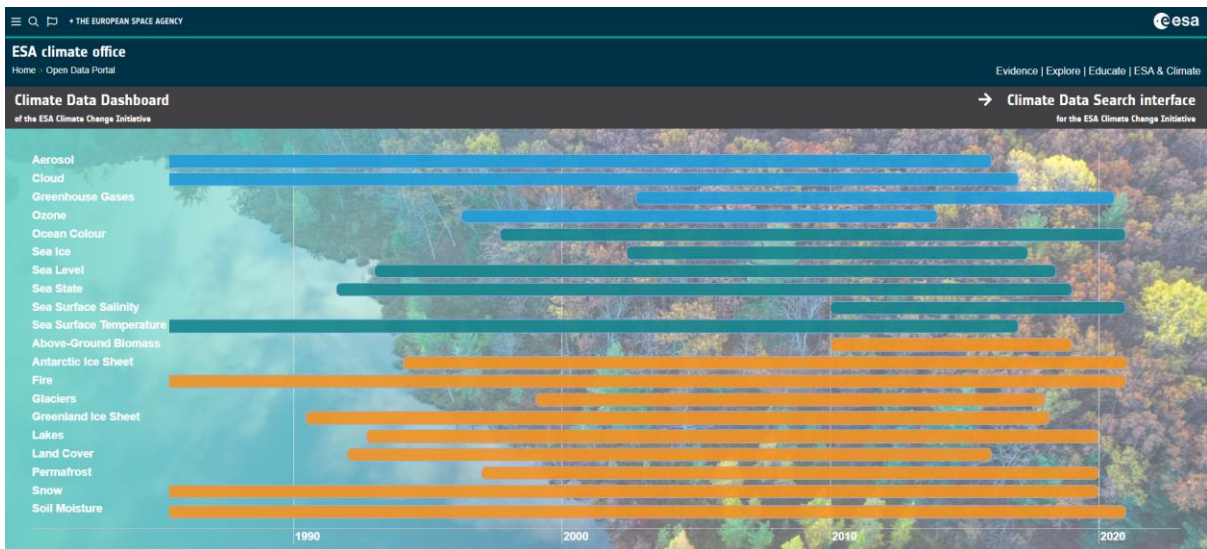


Figure 8: CCI Open Data Portal.

You will be directed to the different data sources of the ECV aerosol. For this exercise, we will select the *Level 3 aerosol Products from Multi-sensor UV* (the black bar, Figure 9).



Figure 9: Aerosol data sources.

On the right-hand panel, you can access the data in your preferred approach, for example using FTP client software³. In principle, you do not need to download the data manually when you use the option ESA CCI Open Data Portal, which allows you to access the data directly with Cate.

Switching the data store selector to Local Data Sources lists all currently available local data sources. These are the ones downloaded from remote sources or ones that you can create from local data files. Press the Add button to open the Add Local Data

³ There are many open source FTP client tools available free of charge under the GNU Public License

Source dialog that is used to create a new local data source. A data source may be composed of one or more data files that can be stacked together along their time dimension to form a single unique multi-file dataset. At the time of writing, only NetCDF (*.nc) data sources are supported. Pressing the Open button will bring up the Open Local Dataset dialog as shown in Figure 10 below:

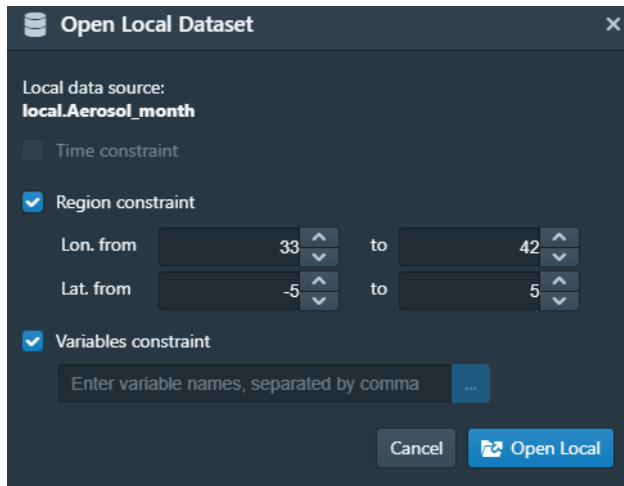


Figure 10: Open local data sources.

CHAPTER SIX

Working with CCI Data

Open and visualise data

We have to download the dataset for you so there is no need to re-download it again. From Cate Data Sources select Local Data Sources and press the plus sign button to add local data source (Aerosolⁱⁱ Climatology), locate the dataset folder and fill in a local name (Figure 11).

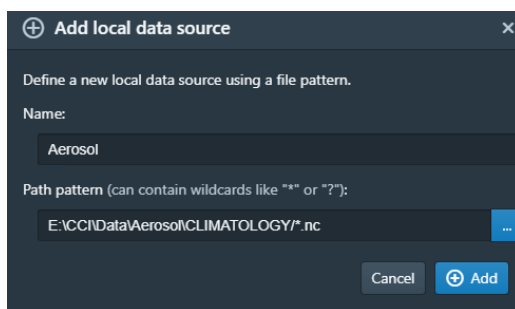


Figure 11: Adding local data source.

Now the aerosol climatology data will be loaded. Monthly climatology indicates data that have been averaged in time such that each month represents the mean over the years of observation. Table 1 shows an example of 10 years of observations in one location and how monthly climatology is estimated. For one location, the operation reduces a 12 (number of months in a year) x 10 (years of observations) matrix to 12 values (most right column in Table 1). Similarly, one can construct daily and seasonal climatology.

Table 1: Example of computing monthly climatology for one location.

Months in a year	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Monthly climatology
Jan											Sum of Jan. / 10
Feb											Sum of Feb. / 10
Mar											Sum of Mar. / 10
Apr											Sum of Apr. / 10
May											Sum of May. / 10
Jun											Sum of Jun. / 10
Jul											Sum of Jul. / 10
Aug											Sum of Aug. / 10
Sep											Sum of Sep. / 10
Oct											Sum of Oct. / 10
Nov											Sum of Nov. / 10
Dec											Sum of Dec. / 10

On the right-panel (red box in Figure 12) a list of variables contained in this data source. You can choose a variable and it will be directly visualised on the globe. You can explore any region on earth by rotating the globe. With the mouse hoover and right-click to place a point marker (Figure 12).

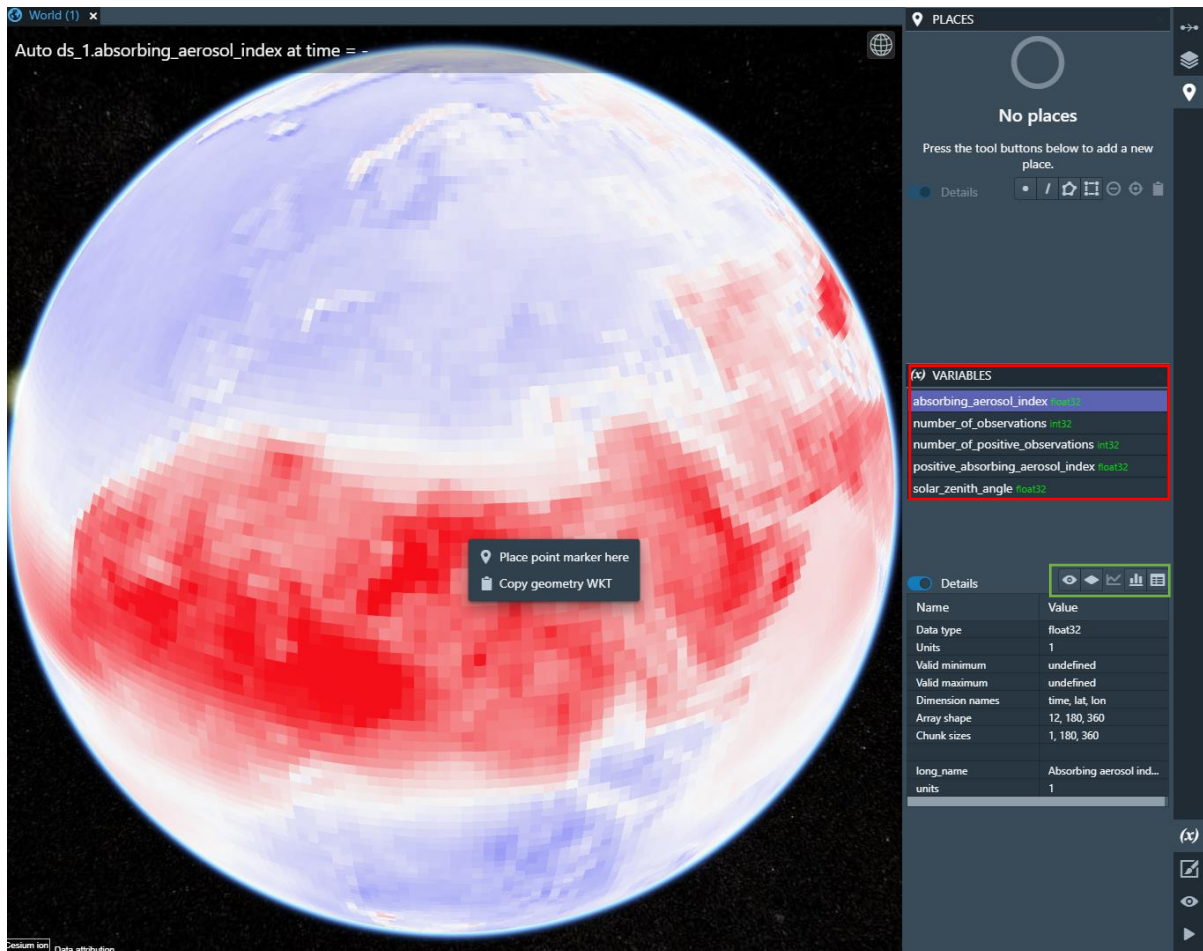


Figure 12: Visualising with CATE and placing a point marker.

From the right panel (green box in Figure 12), create a time series plot from the place marker: select the placemark (a green box surrounding the selected placemark will appear) and then click on the icon to create a time series plot from the selected placemark (Figure 13).

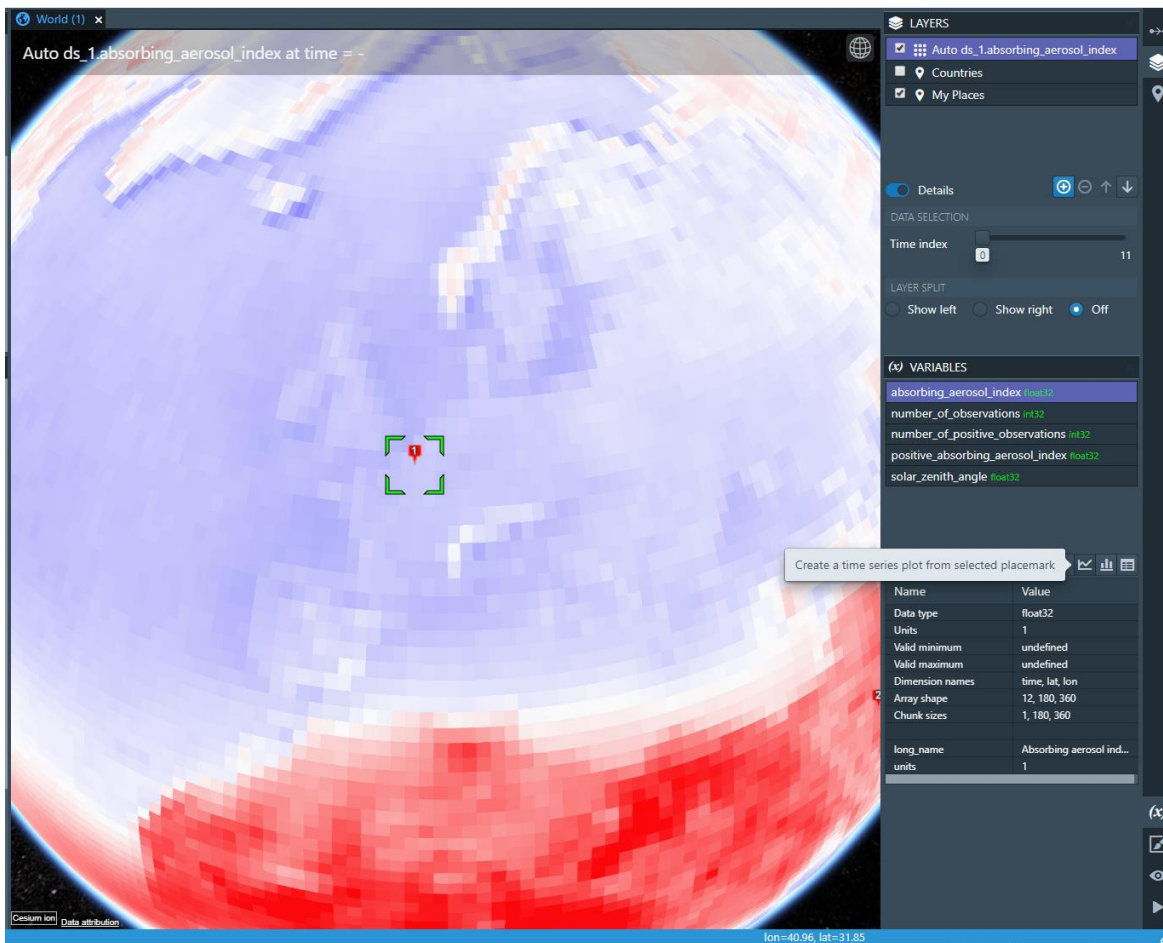


Figure 13: Create a time series plot from the selected placemark.

Hovering the mouse above the icons will indicate their functionality:



Toggle image layer visibility



Add new image layer



Create a time series plot from selected placemark



Create a histogram plot



Show data in table (only first 10.000 values)

A new tab will appear with the time series plot (Figure 14).

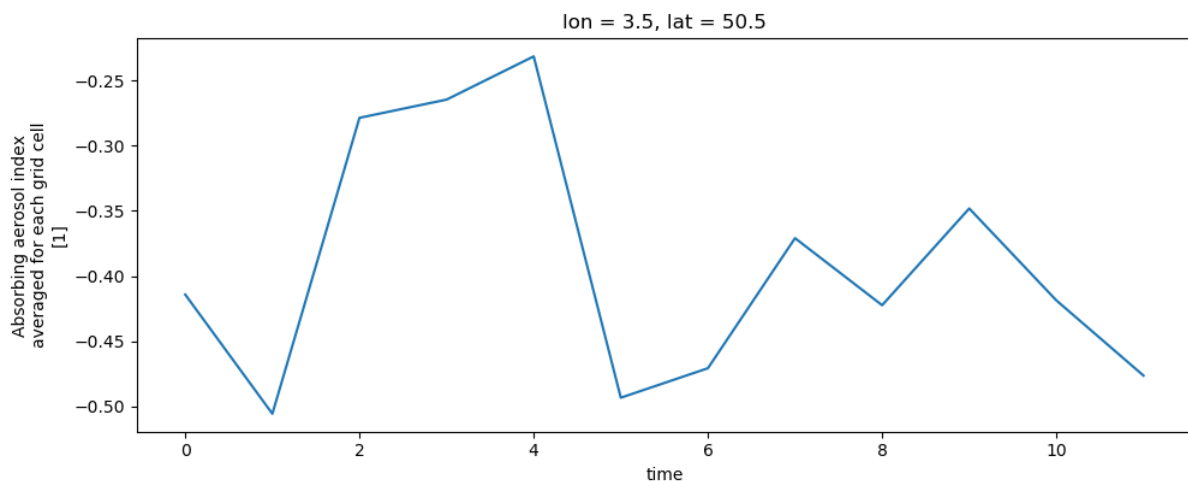


Figure 14: Time series at the place marker 1 of Figure 13.

You can compare with a different region of the world, by placing another place marker, an example is shown here to the right. A new tab will appear with the time series plot of the second place marker will appear.

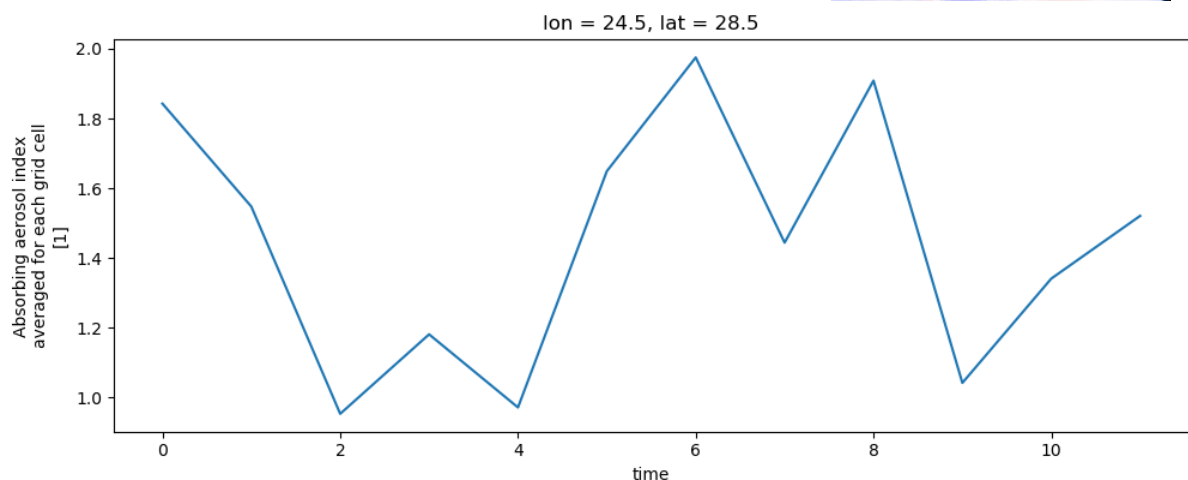
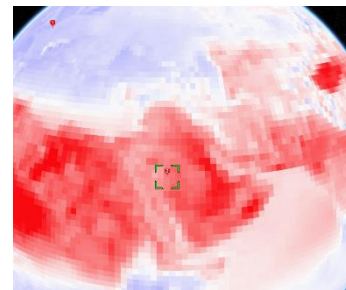


Figure 15: Time series at the second place marker.

There are three main features to learn from this example:

- 1- Each location of climatology data shows 12 values representing the months of the year, with each month being the average of its values of all years of observation.
- 2- The Absorbing Aerosol Index (AAI) is a qualitative quantity that shows the presence of absorbing aerosols. In this case, the ultraviolet (UV) spectral range is used to detect the presence of UV-absorbing aerosols in the atmosphere. Positive values indicate the presence of absorbing aerosol like smoke and dust while small or negative values represent non-absorbing aerosols and clouds.

The values of AAI are dependant on aerosol properties but also on sensor characteristics, and retrieval conditions (solar zenith angle and surface albedo).

- 3- Different locations will have different values and time series. For example, comparing the time series in Figure 14 and Figure 15, we can deduce that in Europe (place marker 1), scattering aerosol is present.

To appreciate the presented time series, you can plot the histogram of the global climatology values of AAI (Figure 16). Now one can immediately notice that the times series of Figure 14 and Figure 15 represent extreme values. For example, the values of Figure 14 are located at the left-end of the histogram, whereas the values of Figure 15 are located at the right-end of the histogram.

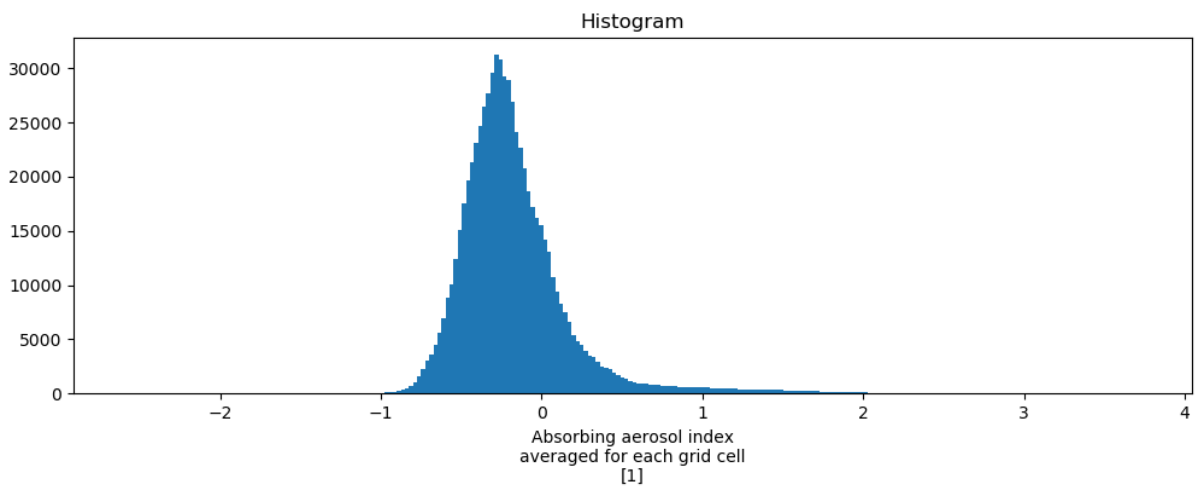


Figure 16: Histogram of the global climatology values of absorbing aerosol index.

Processing data

Operations as used in the Cate context, includes functions that

- read datasets from files;
- manipulate these datasets;
- plot datasets;
- write datasets to files.

From the operation panel, you can search for processing steps and their explanation (Figure 17).

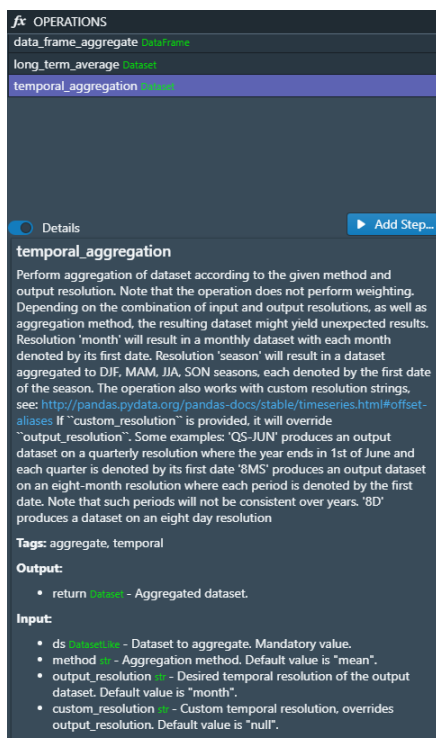


Figure 17: Operations panel.

Let us try another data set on soil moistureⁱⁱⁱ, load the daily data of soil moisture over the last 40 years. Perform a spatial subset to include Kenya (a country that suffers from the effect of climate change on its food security). The bounding box is longitude [33, 42], latitude [-5, 5] as shown in Figure 18.

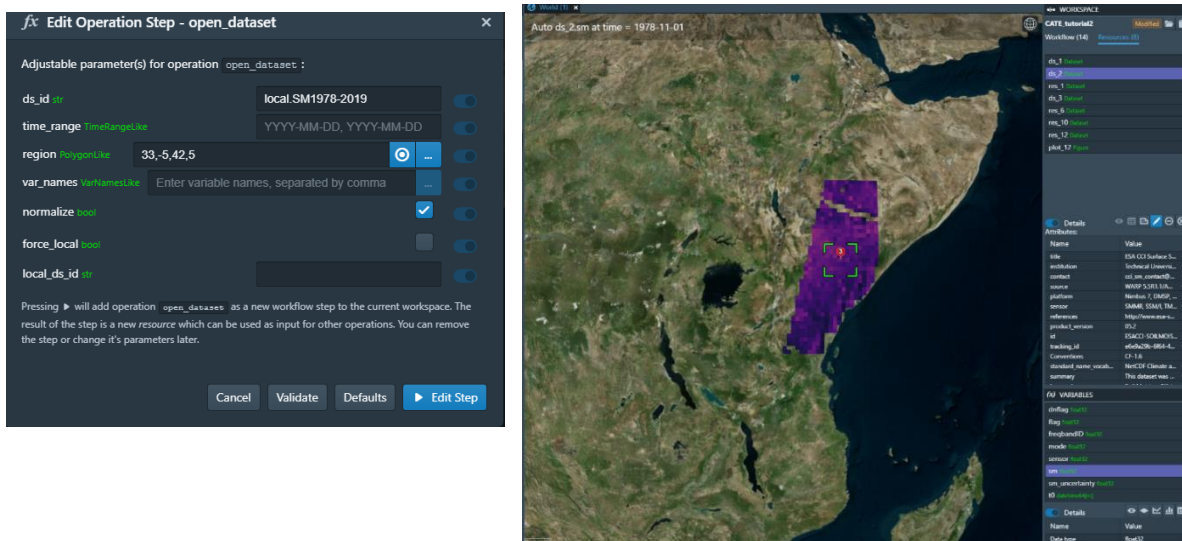


Figure 18: Spatial subset on soil moisture data.

Choose a place marker and plot the time series (Figure 19). You may notice that they are data gaps, let us average to monthly data.

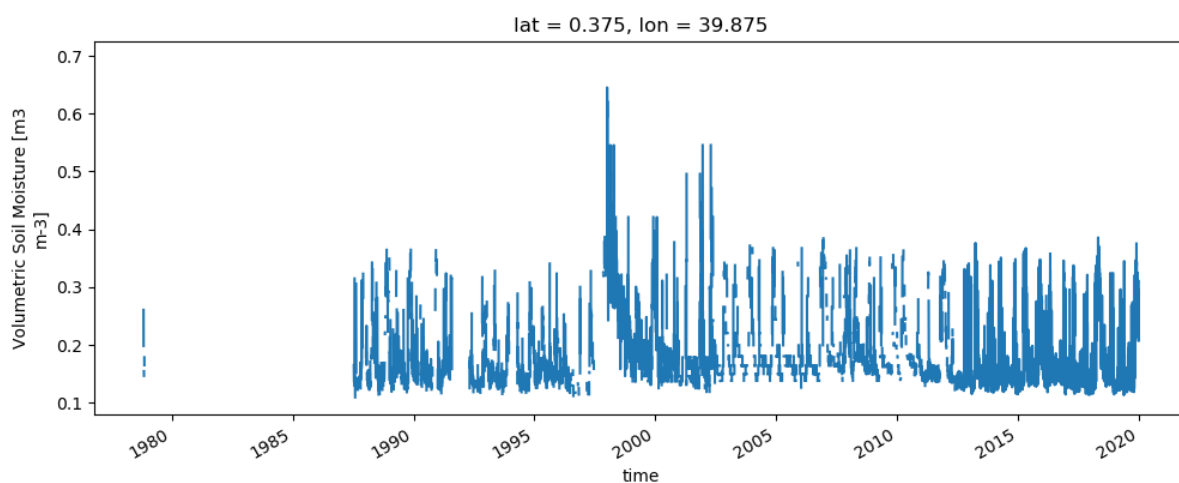


Figure 19: Daily values of soil moisture.

You can remove these gaps by averaging the data according to the month using the operation `temporal_aggregation`. Go to the OPERATIONS panel (Figure 20), and in the search box search for `temporal_aggregation`, and add this processing step. This will bring up a dialog that lets you enter the operation's parameter values (Figure 20). Select the data set of monthly AAI, set the method to mean, and the output resolution to season.

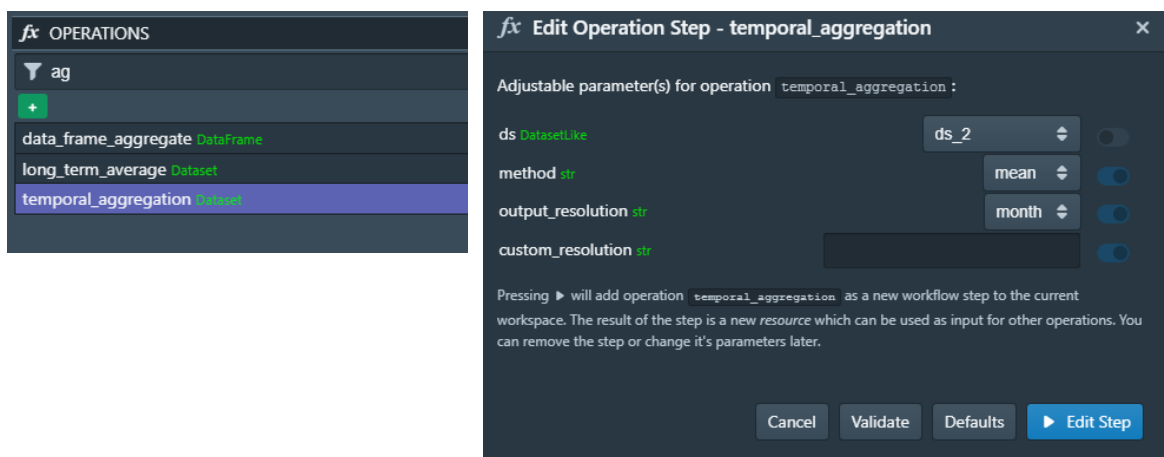


Figure 20: Invoke an operation and fill its parameters.

The results will show data gaps filled in space (Figure 21-a) and time (Figure 21-b).

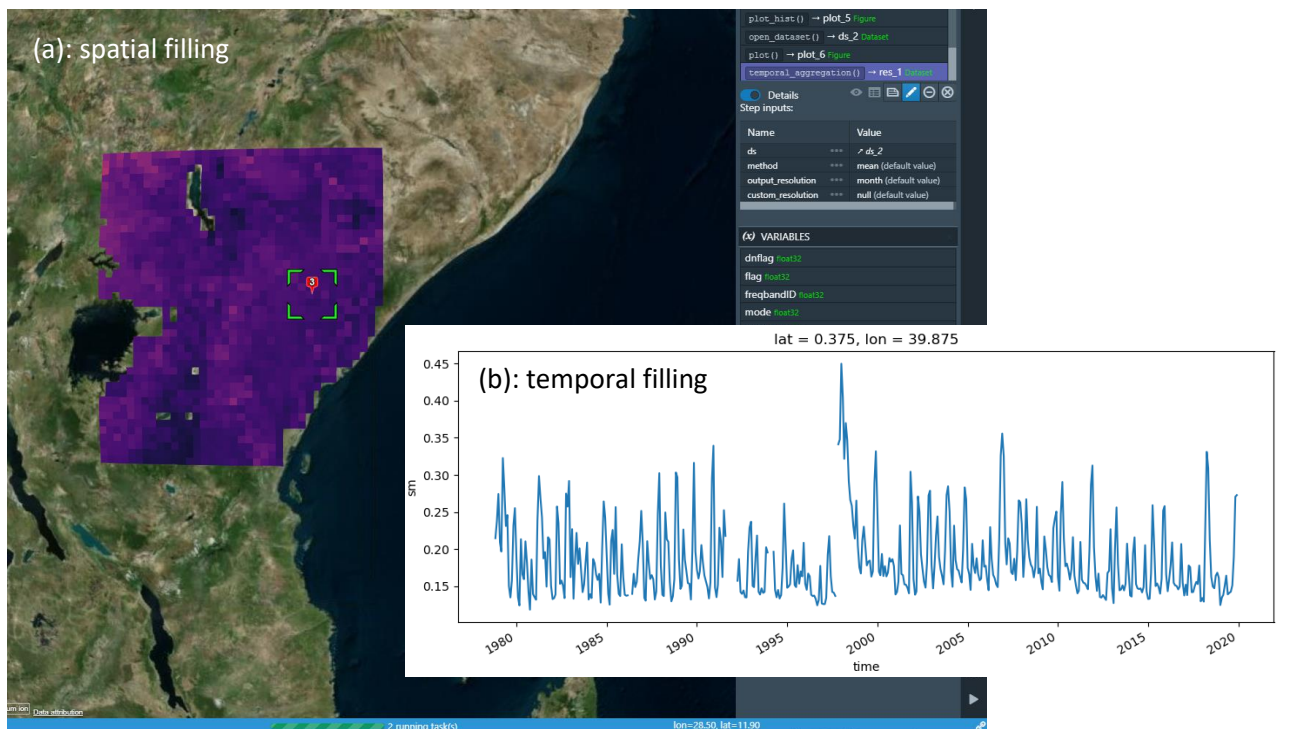


Figure 21: Temporal averaging of daily to monthly data products and the resulting a: spatial filling and (b) temporal filling.

You can notice from Figure 21: Temporal averaging of daily to monthly data products and the resulting a: spatial filling and (b) temporal filling.-b, some reoccurring spikes with the largest one around the year 1998, which correspond to the flooding event in East Africa of this year. Drought events can also be noticed, but are harder to detect visually. A straightforward approach to estimate excesses and deficiencies is to compute the deviation of a value from the historical mean; this is commonly referred to as anomalies. Now Figure 22 clearly shows how much the monthly values are below or above the mean value of soil moisture in the last 40 years.

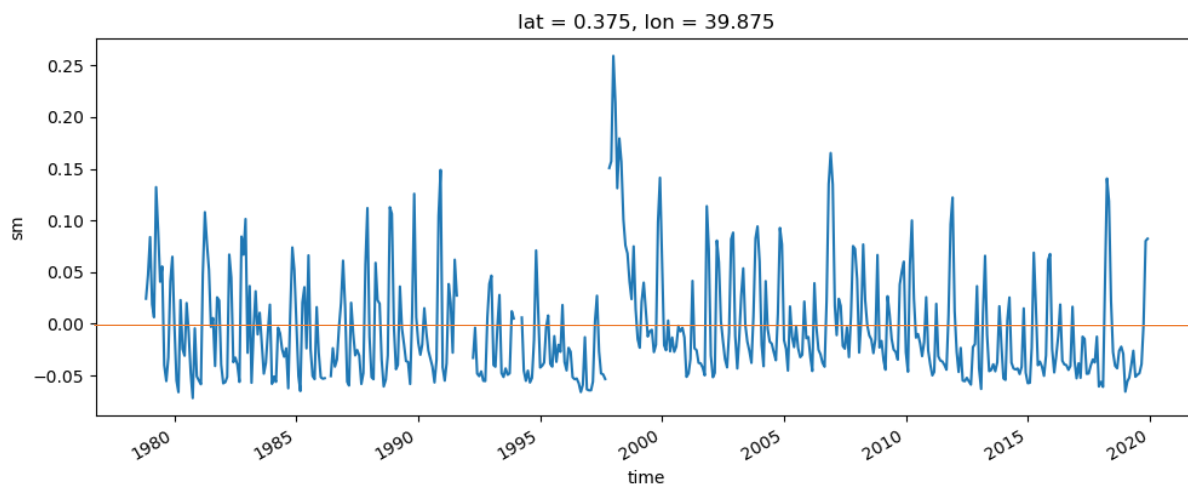


Figure 22: Monthly anomalies of soil moisture at one point.

Monthly climatology can also be computed to investigate the monthly variability using the operation `long_term_average` (Figure 23). The operation takes the data (daily or monthly) and returns the (daily or monthly) climatology (Figure 24).

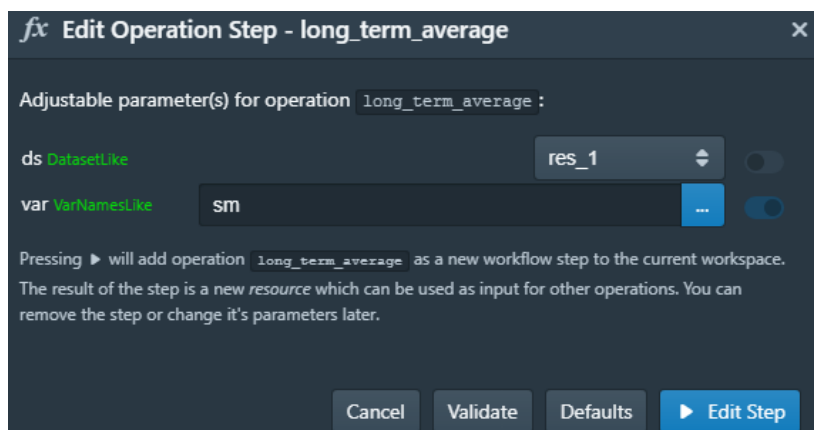


Figure 23: Computing climatology

From this monthly climatology data, the soil moisture values experience two peaks from October to January and from March to June, with an extended “dry” period of lower soil moisture values between June and October.

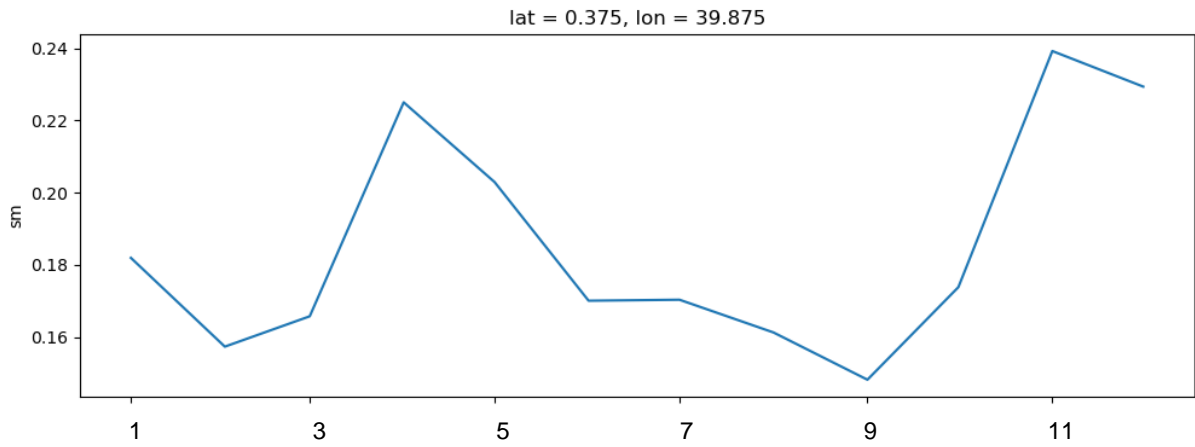


Figure 24: Monthly Climatology of soil moisture.

CHAPTER SEVEN

Accessing Cate functionality from within Jupyter Notebooks

The Cate toolbox is [build upon a python module](#), which can also directly be used from within python. In particular, this allows Cate functionality to be called from so Jupyter Notebooks which can be run and then viewed from any capable browser.

In this section we'll introduce Jupyter Notebooks and show you how to install and run a notebook on your computer.

What are Jupyter Notebooks?

A notebook integrates code and its output into a single document that combines visualizations, narrative text, mathematical equations, and other rich media and separates them in so-called cells. In other words: it's a single document where you can run code, display the output, and also add explanations, formulas, charts, and make your work more transparent, understandable, repeatable, and shareable.

Jupyter Notebooks and dashboard are web apps, and Jupyter can be started as a local Python server to serve these apps to your web browser, making it essentially platform-independent and opening the door to easier sharing of your code developed on the web.

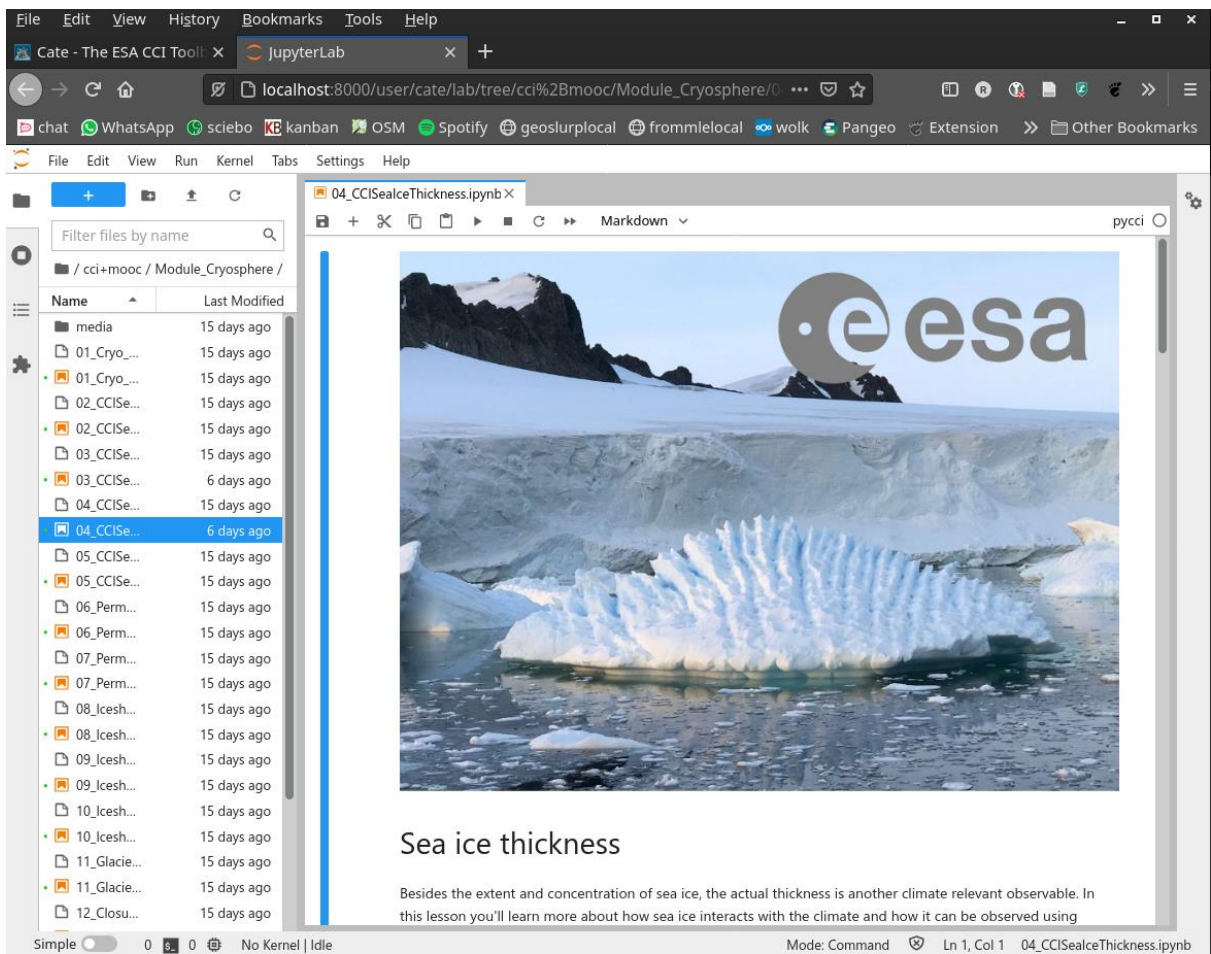


Figure 25: The Jupyter Notebook dashboard can be viewed and controlled from a web browser. The url mentions "localhost" which means that this particular Jupyter Notebook instance is running on the same computer as the browser.

The dashboard's interface is mostly self-explanatory. Each notebook file has an .ipynb extension. And if you would look into this file you would find that it is a text file that describes the contents of your notebook in a format called JSON. Each cell and its contents, including image attachments that have been converted into strings of text, is listed therein along with some metadata.

There are two items appearing in the main menu that you should notice: cells and kernels. These are key both to understanding Jupyter and to what makes it more than just a word processor.

A kernel is a "computational engine" that executes the code contained in a notebook document. A cell is a container for text to be displayed in the notebook or code to be executed by the notebook's kernel. Cells form the body of a notebook. There are two main cell types:

- A *Code* cell contains code to be executed in the kernel. When the code is run, the notebook displays the output below the code cell that generated it.
- A *Markdown* cell contains text formatted using Markdown and displays its output in-place when the Markdown cell is run. Your notebook essentially takes

the Markdown code and converts it into a snippet of html which your browser know how to display.

You can always tell the difference between Code and Markdown cells because code cells have that label on the left and Markdown cells do not. The “In” part of the label is simply short for “Input,” while the label number indicates when the cell was executed on the kernel. In a Jupyter Notebook, there is always one “active” cell highlighted with a border whose color denotes its current mode:

- Green outline — cell is in "edit mode"
- Blue outline — cell is in "command mode"

To run a cell, press the Shift + Enter keys simultaneously, or select the cell and press the triangular play button on the dashboard. There are plenty of other commands we can use. The best way to use them is with keyboard shortcuts. Keyboard shortcuts are a very popular aspect of the Jupyter environment because they facilitate a speedy cell-based workflow. Many of these are actions you can carry out on the active cell when it's in command mode.

Below, you'll find a list of some of Jupyter's keyboard shortcuts. You don't need to memorize them all immediately, but this list should give you a good idea of what's possible.

- Toggle between edit and command mode with Esc and Enter, respectively.
- Once in command mode:
 - Scroll up and down your cells with your Up and Down keys.
 - Press 'a' or 'b' to insert a new cell above or below the active cell.
 - 'm' will transform the active cell to a Markdown cell.
 - 'y' will set the active cell to a code cell.
 - 'd' + 'd' (d twice) will delete the active cell.
 - 'z' will undo cell deletion.
 - Hold Shift and press Up or Down to select multiple cells at once. With multiple cells selected, Shift + 'm' will merge your selection.
- Ctrl + Shift + -, in edit mode, will split the active cell at the cursor. If your keyboard has more than one key corresponding to -, and one zooms out instead of splitting cells, try using the other key.
- You can also click and Shift + Click in the margin to the left of your cells to select them.

From the “Help” menu also additional information about the various keyboard shortcuts can be obtained.

Markdown is a lightweight and popular Markup language which is a writing standard for data scientists and analysts. It is often converted into the corresponding HTML by the Markdown processor which allows it to be displayed in browsers and apps which understand html. One of the advantages of Markdown is that it's plain text, and that, even when it's not converted to html is still easily understandable and human readable.

For additional information on Markdown, see:

<https://www.datacamp.com/community/tutorials/markdown-in-jupyter-notebook>.

Behind every notebook runs a kernel. When you run a code cell, that code is executed within this kernel. Any output is returned back to the cell to be displayed. The kernel's state persists over time and between cells — it pertains to the document as a whole and not individual cells. This also means that the order of executing of cells matter!

Most of the time when you create a notebook, the flow will be top-to-bottom. But it's common to go back to make changes. When we do need to make changes to an earlier cell, the order of execution we can see on the left of each cell, such as 'In [6]' (in front of a cell), can help diagnose problems by seeing what order the cells have run in.

And if we ever wish to reset things, there are several useful options from the Kernel menu:

Restart: restarts the kernel, thus clearing all the variables, *etc.* that were defined.

Restart & Clear Output: same as above but will also wipe the output displayed below your code cells.

Restart & Run All: same as above but will also run all your cells in order from first to last.

If your kernel is ever stuck on a computation and you wish to stop it, you can choose the Interrupt option.

Depending on the type of installation, Jupyter provides the option to change the kernel – programming languages. Many other languages, in addition to Python, may be used in the notebook. By creating a new notebook from the dashboard menu, by selecting Python, you are actually choosing which kernel to use. Note that, in this tutorial, we only consider Python notebooks.

Installing Jupyter

Depending on the system your running, [installing Jupyter may be slightly different](#). But it essentially consists of (1) installing a Python interpreter, (2) installing the Jupyter package and (3) installing additional requirements which may be needed to run your specific notebooks.

Here we focus on installing a Jupyter Notebook by using [Anaconda](#) which is a self contained software package manager targeting data scientists. Installing the default Anaconda package automatically includes an up to date Python interpreter and a Jupyter server (steps 1 & 2 above). Anaconda installs many packages, and you do probably not need all of those. So if you want to have a smaller or customised installation you can take the somewhat longer road and [install jupyter through installing a python interpreter and the pip tool](#).

Step 1 Install Jupyter using Anaconda

Download and run the Anaconda installer by visiting <https://www.anaconda.com/products/individual> and choosing the appropriate packages matching your operating system.

Step 2 Install Cate and its requirements

To run functionality of the Cate python module we need to install additional packages, besides Cate itself, such as for [xarray](#) and [geopandas](#). We will download a text file (in yaml format) from <https://raw.githubusercontent.com/CCI-Tools/Cate-edu/main/binder/environment.yml> which contains everything which is needed to run Cate and tell Anaconda to install the associated packages.

To do this start an Anaconda prompt (in windows: through start), and execute the following to download the file:

```
curl -o environment.yml https://raw.githubusercontent.com/CCI-Tools/Cate-edu/main/binder/environment.yml
```

and then

```
conda env update --file environment.yml
```

to install the required packages

Step 3 Start a Jupyter Notebook

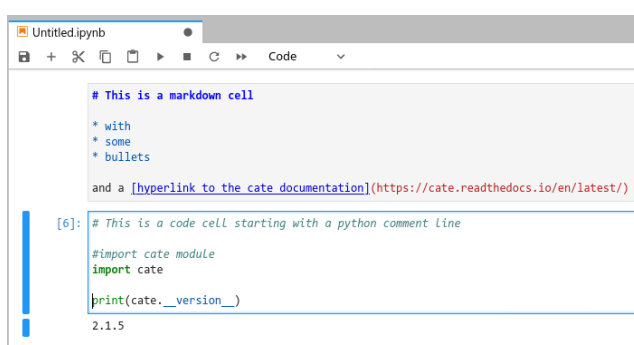
Once everything is installed you can open an Anaconda prompt and type

```
jupyter notebook
```

If everything goes well, this will automatically open up your default browser and point you to the notebook instance. If your browser doesn't open automatically, keep an eye on the output in the prompt and copy and paste the url, including the token in your browser's address bar.

Step 4 Create and try out try out your first notebook

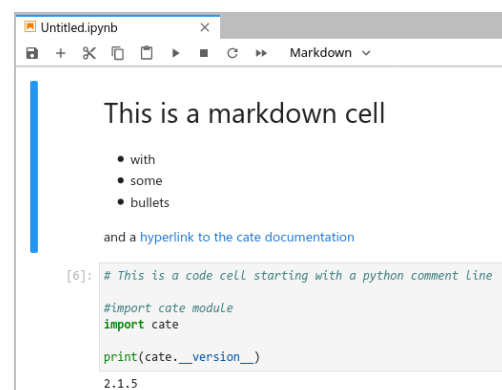
Once you have the Jupyter dashboard up in your browser, you can create a new notebook and test the installation. By creating a markdown cell, and a code cell.



The screenshot shows a Jupyter Notebook interface with a code cell selected. The code cell contains a Python comment, a markdown cell reference, and a code cell reference. The output of the code cell is displayed below the code.

```
# This is a markdown cell
* with
* some
* bullets
and a [hyperlink to the cate documentation](https://cate.readthedocs.io/en/latest/)

[6]: # This is a code cell starting with a python comment Line
#import cate module
import cate
print(cate.__version__)
2.1.5
```



The screenshot shows a Jupyter Notebook interface with a markdown cell selected. The markdown cell contains a title, a bulleted list, and a hyperlink. The output of the code cell is displayed below the markdown cell.

```
This is a markdown cell

* with
* some
* bullets

and a hyperlink to the cate documentation

[6]: # This is a code cell starting with a python comment Line
#import cate module
import cate
print(cate.__version__)
2.1.5
```

References

Cate user Manual: https://Cate.readthedocs.io/en/latest/user_manual.html

Cate operations: ESA CCI_Reference document on main CATE operations.PDF

Notes

ⁱ **What is Aerosol?** Tiny particles suspended in the atmosphere originating from natural processes (e.g. dust storm, volcano ashes) and human emissions (e.g. from fossil-fuel burning and agricultural activities). Aerosol interact with incoming radiation from the sun. Based on their types, aerosol particles could either cool or warm the atmosphere. Near the surface, these fine aerosol particles are a major concern for public health, causing cancer and cardiopulmonary mortality

ⁱⁱ

ⁱⁱ (as above)

ⁱⁱⁱⁱⁱⁱ **What is soil moisture?** Water molecules in the root zone layer of the soil that are available for plants. Soil moisture influences hydrological and agricultural processes, run-off generation, drought development and many other processes. Severe drought, i.e. deficiency in soil moisture, threaten plants production and may cause animals and habitats loss.