Gravity Recovery and Climate Experiment (GRACE) Groundwater Subsetting Tool

Training Manual

UNITED NATIONS
الاسكوا
ESCWA

Shared Prosperity Dignified Life

Sweden
Sverige

UNITED NATIONS
الاسكوا
ESCWA

Shared Prosperity **Dignified Life**

VISION

ESCWA, an innovative catalyst for a stable, just and flourishing Arab region

MISSION

Committed to the 2030 Agenda, ESCWA's passionate team produces innovative knowledge, fosters regional consensus and delivers transformational policy advice. Together, we work for a sustainable future for all.

E/ESCWA/CL1.CCS/2024/Manual.1

Economic and Social Commission for Western Asia

Gravity Recovery and Climate Experiment (GRACE) Groundwater Subsetting Tool

Training Manual

Authored by: Norman L. Jones, Civil and Construction Engineering, Brigham Young University Provo, Utah, United States of America.

Reviewed by: Ziad Khayat and Tracy Zaarour, Economic and Social Commission for Western Asia.

United Nations publication issued by ESCWA, United Nations House,

Riad El Solh Square, P.O. Box: 11-8575, Beirut, Lebanon.

Website: www.unescwa.org.

Acknowledgements

All screenshots in this manual were taken from the GRACE Groundwater Subsetting Tool (GGST) that was developed at the Hydroinformatics Laboratory at Brigham Young University in Provo, Utah, United States of America. GGST and the associated tools were originally developed via funding from the National Aeronautics and Space Administration: 80NSSC20K0155 and the United States Agency for International Development: Cooperative Agreement with SERVIR West Africa Hub.

Contents

List of tables

List of figures

Introduction

Groundwater is a crucial pillar of water security in the Arab region as it is the primary source of freshwater in more than half of the Arab States, and therefore heavily relied upon. Recognizing the importance of this resource, the United Nations Economic and Social Commission for Western Asia (ESCWA) has secured funding from the United Nations Development Account and the Government of Sweden to support Arab States in improving their water security by strengthening their capacities for the sustainable management of groundwater resources.

These activities aim to: (1) improve the availability and accessibility of groundwater data and information through the Arab Groundwater Knowledge Platform; (2) improve the assessment of climate change impact on groundwater resources in the Arab region through capacity development and pilot case studies; and (3) advance the use of innovative technologies for the management of groundwater resources at national and transboundary levels. This training manual was developed to build capacities related to the use of the Gravity Recovery and Climate Experiment (GRACE) mission to monitor groundwater storage change (Tapley and Reigber, 2002).

Since the launch of the National Aeronautics and Space Administration (NASA) GRACE mission in 2002, it has been possible to monitor groundwater storage changes at a large scale using monthly estimates of total water storage anomalies in equivalent water height. The original mission ended in 2017 and was succeeded by the GRACE Follow-on mission (GRACE-FO) in 2018, which continues to provide large-scale estimates of total water storage anomalies.

While various tools have been developed for processing and visualizing GRACE data, the GRACE Groundwater Subsetting Tool (GGST) is specifically crafted to assist regional stakeholders and decision makers in groundwater resource management (McStraw et al., 2021). GGST processes raw GRACE data to eliminate anomalies and enhance resolution, aiming to support the identification and characterization of long-term groundwater storage changes in selected regions. Particularly useful in data-poor areas or regions where trends may be obscured by noise from well data, GGST leverages GRACE mission data to compute and display alterations in water storage through a web-based mapping system, integrating data from both the GRACE and GRACE-FO missions.

GGST uses NASA Global Land Data Assimilation System (GLDAS) surface water data to derive groundwater storage changes. It accepts shapefiles to define regions representing countries, basins or aquifers. It then aggregates the water volume changes in those regions and displays the results as time series plots for the whole region or at selected points. It also displays an animated map of the storage change anomalies. Example applications of GGST in a regional context have been conducted in Niger (Barbosa et al., 2022)and elsewhere.

Overview

1

1. Overview

The GGST app uses GRACE data to generate time series and animated maps of groundwater storage changes. GRACE provides monthly estimates of water storage anomalies in equivalent water height and has provided monthly gravity field solutions since April 2002. Estimates of mass variability and associated observational errors are available on a global 300 km grid. GRACE has proved an effective tool for characterizing groundwater storage changes in large regions (Famiglietti et al., 2011; Rodell and Famiglietti, 2002; Syed et al., 2009), including in regions in the Middle East (Voss et al., .2013)

While several tools have been developed for processing and visualizing GRACE data, GGST is designed specifically to support groundwater resource management by regional stakeholders and decision makers by carefully processing the raw GRACE data to remove anomalies and improve resolution. This is achieved by separating the groundwater component from the other water storage components using GLDAS, subsetting the data to specific regions of interest and by presenting the results in a simple, intuitive interface. The algorithm used to process the GRACE and GLDAS data to produce groundwater anomalies on both a global and regional scale is described in detail in chapter two.

Users can access GGST using the Tethys web application or by using an application programming interface (API) and the associated Google Colaboratory notebook that makes API intuitive to use. A brief introduction to these two methods is provided – section 1.B.

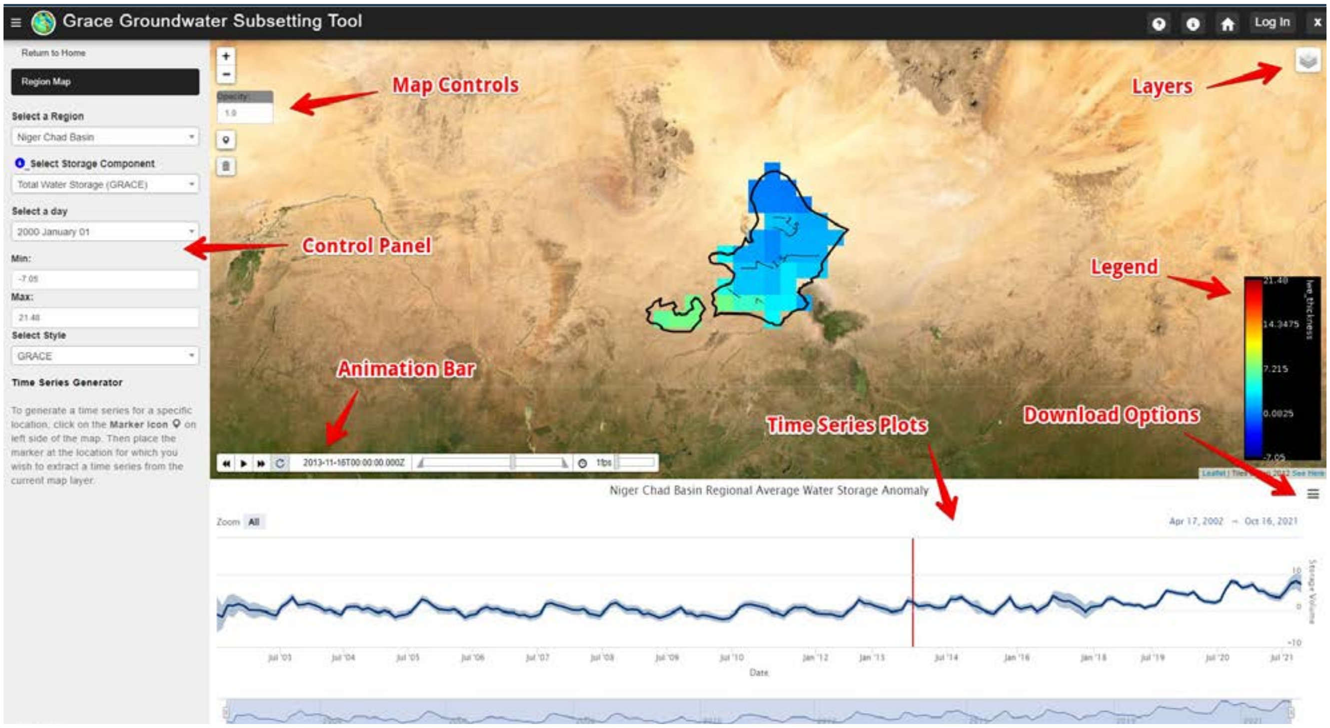
A. Groundwater Subsetting Tool web application

The GGST web application was built using the Tethys platform. Tethys is a web-based application development framework for the rapid deployment of end-user-focused tools that follow modern, consistent, scalable, cross-platform, reusable, web programming paradigms (Jones et al., 2014; Swain et al., 2016). Tethys is built on commonly used web programming frameworks (e.g., Django, GeoServer, PostGIS, OpenLayers). It is an open-source platform that allows anyone to observe and use GGST as a decision support system to ensure sustainable usage of groundwater. It was developed in the Brigham Young University (BYU) Hydroinformatics Laboratory and is now supported by a growing user and developer community. To access the GGST web application, visit https://apps.geoglows.org/apps/ggst/.

Everyone can open the app to view the current uploaded regions and download the time series plots. Figure 1 shows how to manipulate the map and download the data. Users can change the storage component displayed and the colour bar style. Use the animation bar to view the storage change over time. Users can also download the time series plots as an image or as a table. The web app does not yet support downloading the network Common Data Form (netCDF) file raster that is displayed, but this can be downloaded using API.

To upload and delete regions from the Tethys web application, users must log in with an administrator (admin) account. If you have admin access, follow the instructions on the Adding and Delete Regions page. If not, consider using the API method or reach out to BYU or ESCWA's team if you would like a region uploaded.

Figure 1. GRACE Groundwater Subsetting Tool interface screenshot



The GGST app can be accessed in the following locations:

- **SERVIR West Africa Portal**: Official Tethys portal hosted by the SERVIR Science Coordination Office for the West Africa Hub (https://tethyswa.servirglobal.net/apps/).
- **BYU Portal:** A Tethys portal hosted by BYU for experimental web applications developed by the BYU Hydroinformatics Laboratory (https://tethys.byu.edu).
- **GEOGloWS Portal:** A portal associated with the United Nations Global Streamflow and Flood Forecasting System global streamflow forecasting initiative (https://apps.geoglows.org/apps/).
- **ESCWA Cloud:** The GGST app is also available from ESCWA's cloud and is used to process GRACE groundwater data for the Arab Region and to keep the Arab Groundwater Knowledge Platform updated with GRACE data. To process a specific region from the Arab domain, a request can be submitted through https://agwkp.unescwa.org/.

B. Application programming interface and Google Colaboratory notebook

In addition to using the GGST web application through one of the portals noted previously, it is also possible to access the core functionality of the GGST tool through a Python programming language-based API. The advantages of API are that you can retrieve data on a new region of interest without having admin access to the Tethys web application. You can also download a complete zip file of the regions netCDF raster files. You may implement API on your own, but we recommend using the Google Colaboratory notebook hosted on GitHub which is designed to run each of the API functions, and help you download and visualize the data. For more detailed documentation please visit the API page. Google Colaboratory is a web-based Python programming environment hosted in the Google Cloud – a component of the Google Drive environment.

Computational algorithm

2

2. Computational algorithm

The GSST web application relies on the Earth observation data collected by NASA through satellites which map the gravitational field of the Earth. Changes in gravity are driven by changes in water storage, offering a rare opportunity to monitor groundwater level through satellites coupled with estimated surface water.

The GRACE mission was launched in March 2002. It consists of a pair of satellites that are 400 km above the Earth and are separated by 200 km. As the satellites pass over different regions of the Earth, the front and rear satellites are pulled slightly forward and backward in response to subtle changes in the Earth's gravitational field caused by changes in surficial mass. This causes the distance between the satellites to vary, and the changes are recorded by a k-band microwave whose accuracy is within 10 microns. The GRACE satellites follow a varying path that covers the entire Earth about once per month. This data is then processed by NASA to produce a map of the Earth's gravitational field. Each month a new map is generated, and the differences are calculated to produce a gravity anomaly map. The changes in mass are assumed to be primarily caused by the change in water storage. Each month, NASA generates a gridded map of total water storage anomaly at a 3 degree resolution. This map is then downscaled using a mass conservation algorithm to a 0.5 degree resolution and made available for download in the netCDF multidimensional raster format.

A. Derivation of groundwater data set

The groundwater component of the GRACE raw data can be separated using a mass balance approach, with GLDAS models to compute the surface water component of the data. To compute total surface water storage, sum the components of the GLDAS models that represent surface water storage and then subtract this total from the GRACE data set to estimate a groundwater storage anomaly (GWSa) data set.

The GSST application uses four data sets:

- GRACE total water storage anomaly (TWSa).
- GLDAS canopy storage (CAN).
- GLDAS snow water equivalent (SWE).
- GLDAS soil moisture (SM).

To compute GWSa, three components of the GLDAS models are used: CAN, SWE and SM. Each GLDAS component is converted to an anomaly format by subtracting the mean centred on values from 2004 to 2009, and then averaged across the three GLDAS models to produce a component anomaly data set: CANa, SWEa and SMa. The standard deviation from the three GLDAS models is used to help estimate uncertainty.

GLDAS files are downloaded, formatted as netCDF and stored locally. Normally, the data is acquired in a gridded format with a 1 degree latitude by 1 degree longitude resolution, which is then converted to a 0.5 degree resolution. This conversion is performed by an area-weighted average of the four GRACE grid cells coincident with each GLDAS grid cell. The converted files are then used to compute the groundwater anomaly using a mass balance approach. The groundwater anomaly is the difference between TWSa and the sum of the surface water component anomalies.

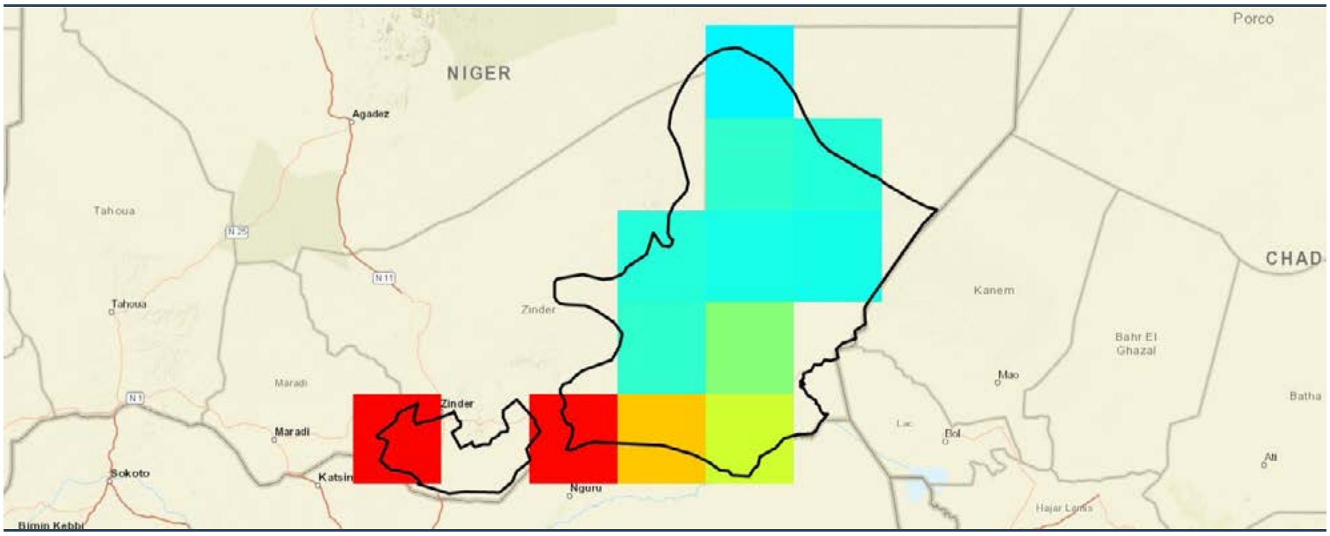
$$GWa = TWSa - (SWEa + CANa + SMa)$$

The result of this computation is GWSa, a tested and approved method to predict long-term changes in groundwater storage.

B. Grid subsetting

For regional subsetting, the user provides a shapefile that defines the boundary of the region of interest. We then select the cells that have centres within the defined boundary and calculate the average storage anomaly for each of the components: TWSa,

Figure 2. Region shapefile uploaded – Chad Basin, Niger screenshot



SWEa, CANa and SMa resulting in a time series from 2002 to the present for each component on a monthly time step. Figure 2 shows the Chad Basin in Niger subsetted and displayed with the region shapefile. For water storage, the average of each component is multiplied by the area of the region, resulting in volume anomalies.

C. Uncertainty estimates

It is critical to understand that the results of these predictions have uncertainties and limitations. To compute the uncertainty of the groundwater storage component, estimates from both GRACE and GLDAS are combined by computing the square root of the sum of the squares of the uncertainty of the individual components as measured by their standard deviations:

$$\sigma GWa = \sqrt{(\sigma TWSa)^2 - (\sigma SWEa)^2 - (\sigma CANa)^2 - (\sigma SMa)^2}$$

The resulting estimates of groundwater data are not suitable for highly precise or localized applications such as the placement of wells; rather, these data serve as an estimate of general trends in groundwater storage.

D. Storage depletion curve

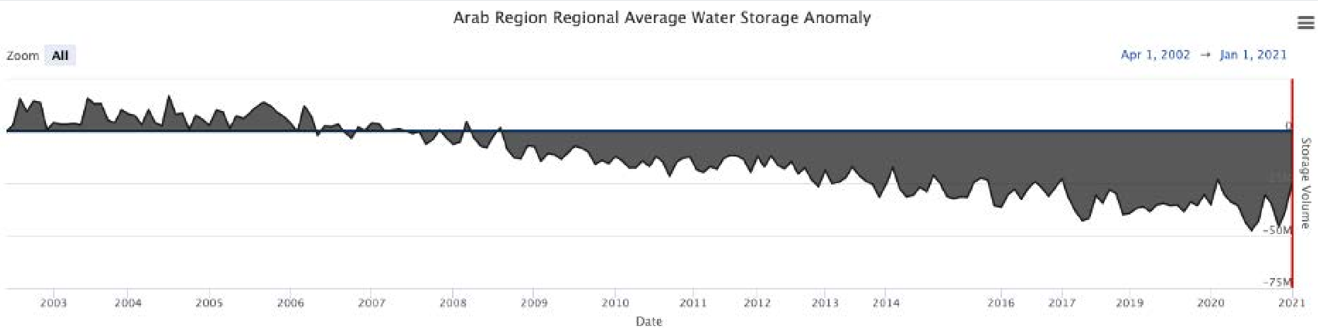
GGST offers an option of viewing time series data in the format of a storage depletion curve, which is the time-integral of the storage anomaly.

The storage depletion curve presents cumulative changes in water component storage relative to levels when the GRACE missions began distributing data in April 2002. The storage depletion curve is used in groundwater management since it offers a simple visualization of how much storage aquifers have gained or lost since a given point in time.

To compute the depletion, GWSa are summed over time to determine changes in groundwater storage volume for the region. These data show if a region is depleting storage or if groundwater is recharging in the region, thereby providing valuable information relative to groundwater sustainability.

Figure 3 is an illustration of the water storage anomaly in northern Africa and the Arabian Peninsula from 2002–2021. It shows that the groundwater in the region has been depleting since early 2009 onward.

Figure 3. Arab Region Regional Average Water Storage Anomaly screenshot



E. Limitations

GRACE data comes with limitations that users need to know and understand. The data is provided at a relatively low resolution (1 degree latitude by 1 degree longitude) representing a 100 km x 100 km square, approximately. At such a low resolution, basing decisions on a single cell comes with high and unknown uncertainties.

Even with these limitations, GRACE data provides valuable insights into aquifers such as regions that are depleting and recharging hence allowing managers to sustainably use their groundwater resources. The best use of GGST is to draw general trends in aquifers rather than selecting a placement of a well.

It is recommended that, whenever possible, the data be validated with local data. GGST displays the uncertainties in the data calculations as error bands on time series, providing context on regions and different time periods.

F. Software availability

The GGST web application was created using Tethys Platform developed in the BYU Hydroinformatics Laboratory. Section 1.A for more information on this web application and links to use it.

Adding and deleting regions 3

3. Adding and deleting regions

This chapter describes how to upload new regions to the GGST app. When a new region is uploaded, it is automatically processed and the storage components – including subsetting netCDF files and storage time series – are computed and stored for visualization. The new region is added to the list of regions for the app, and it can be selected and viewed. This section also describes how to delete regions and associated files.

A. Uploading a region

As described in section 1.A, the GGST web application is hosted on four different Tethys portals.

To upload regions on the app, visit the portal of your choice and log in using the "Log In" link in the upper right corner of the portal window as shown in figure 4.

Without logging in you can see the App navigation pages: Home and Global Map. These allow you to view previously uploaded regions and create time series graphics for any singular point on the globe. Once you log in with administrative privileges, you will see the additional configuration pages: Add a Region, Delete a Region and Update Global Files. Update Global Files is used to download the latest GRACE and GLDAS files from the NASA server.

To add a new region, first prepare a shapefile for the region consisting of four files: *.shp, *.dbf, *.prj and *.shx. The projection for the shapefile should be EPSG:4326 – WGS 84. The four files should not be zipped together. Figures 5 to 9 provide a visual guide.

Figure 4. Tethys Portal log in page screenshot

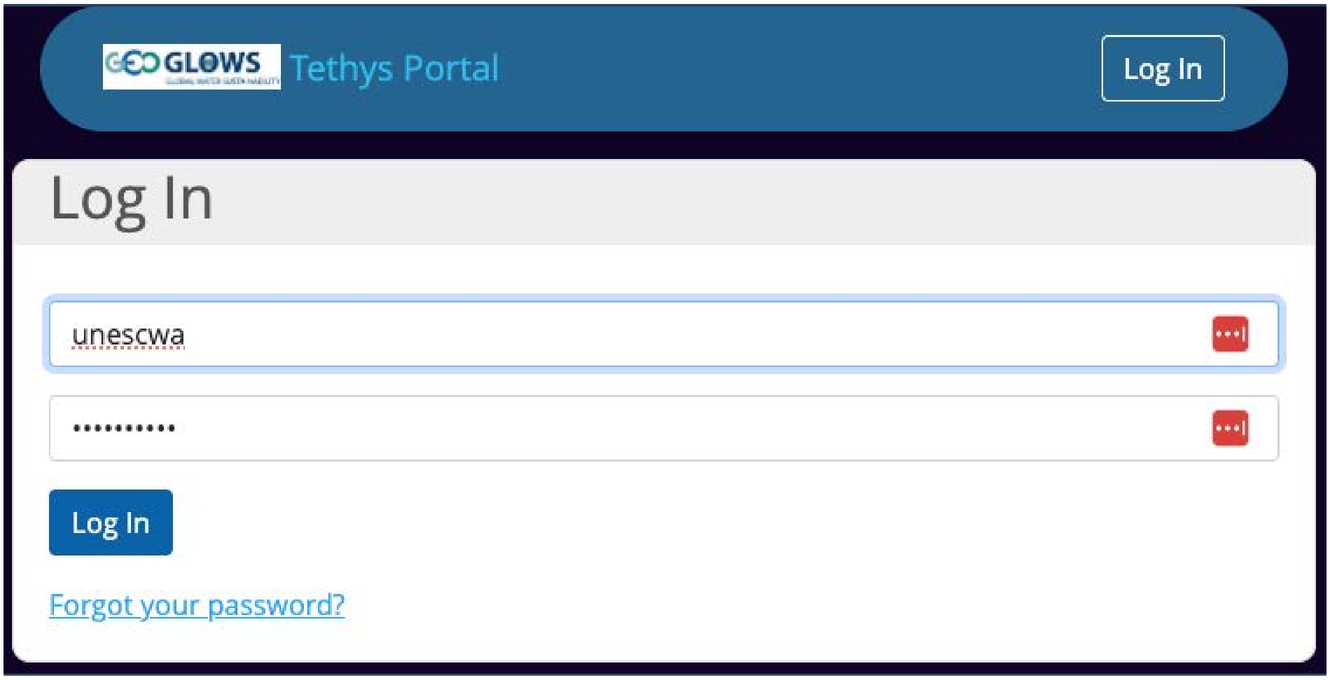


Figure 5. Adding a region in the GRACE Groundwater Subsetting Tool: step 1 screenshot

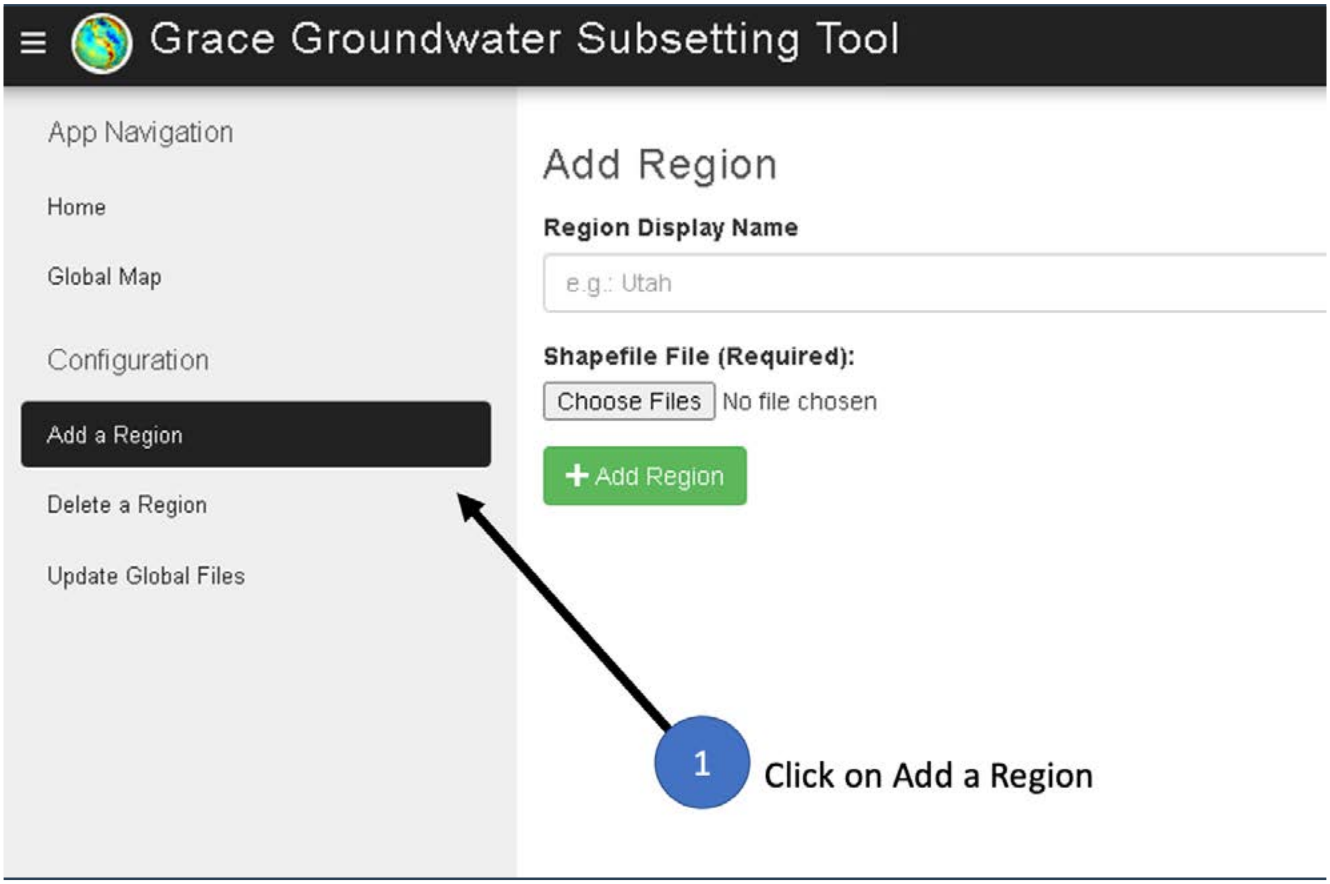


Figure 6. Adding a region in the GRACE Groundwater Subsetting Tool: step 2 screenshot

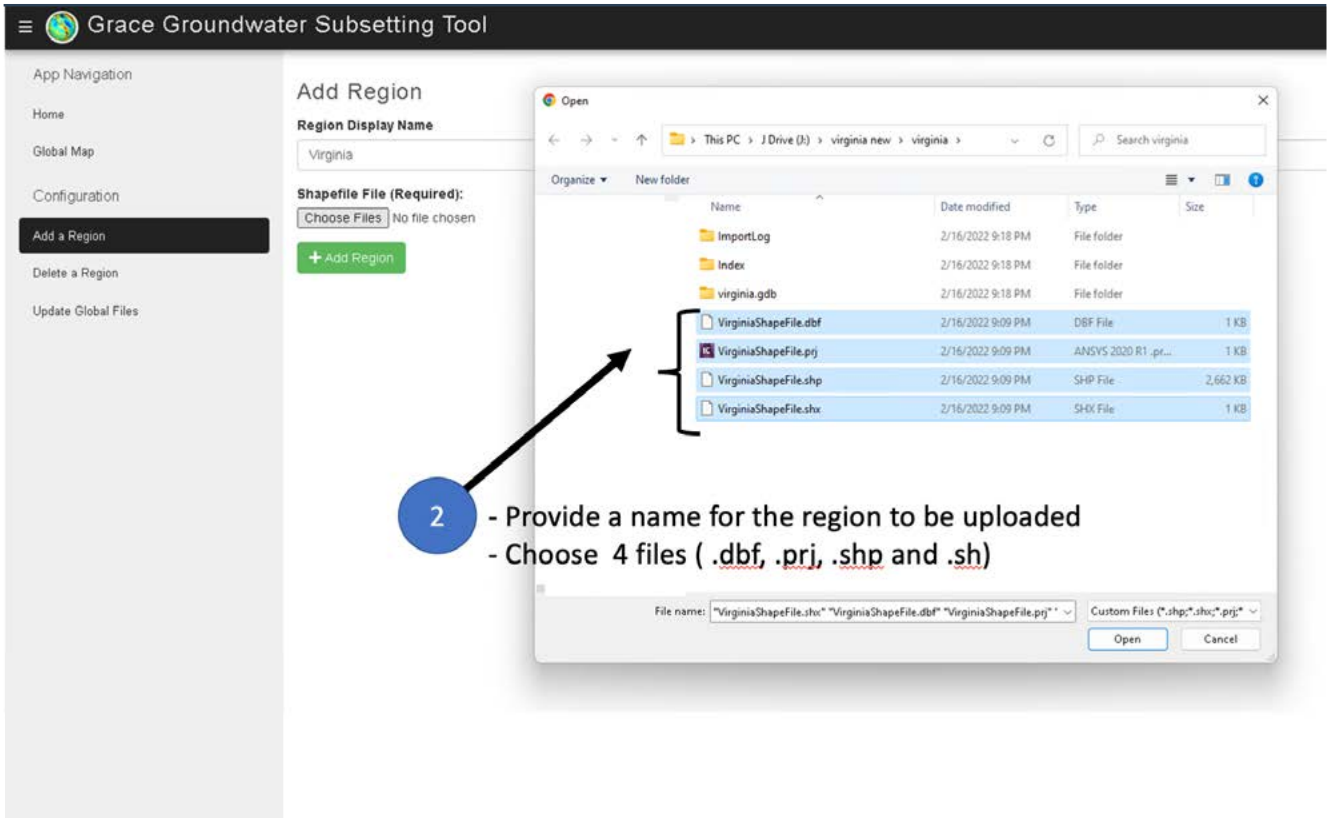


Figure 7. Adding a region in the GRACE Groundwater Subsetting Tool: step 3 screenshot

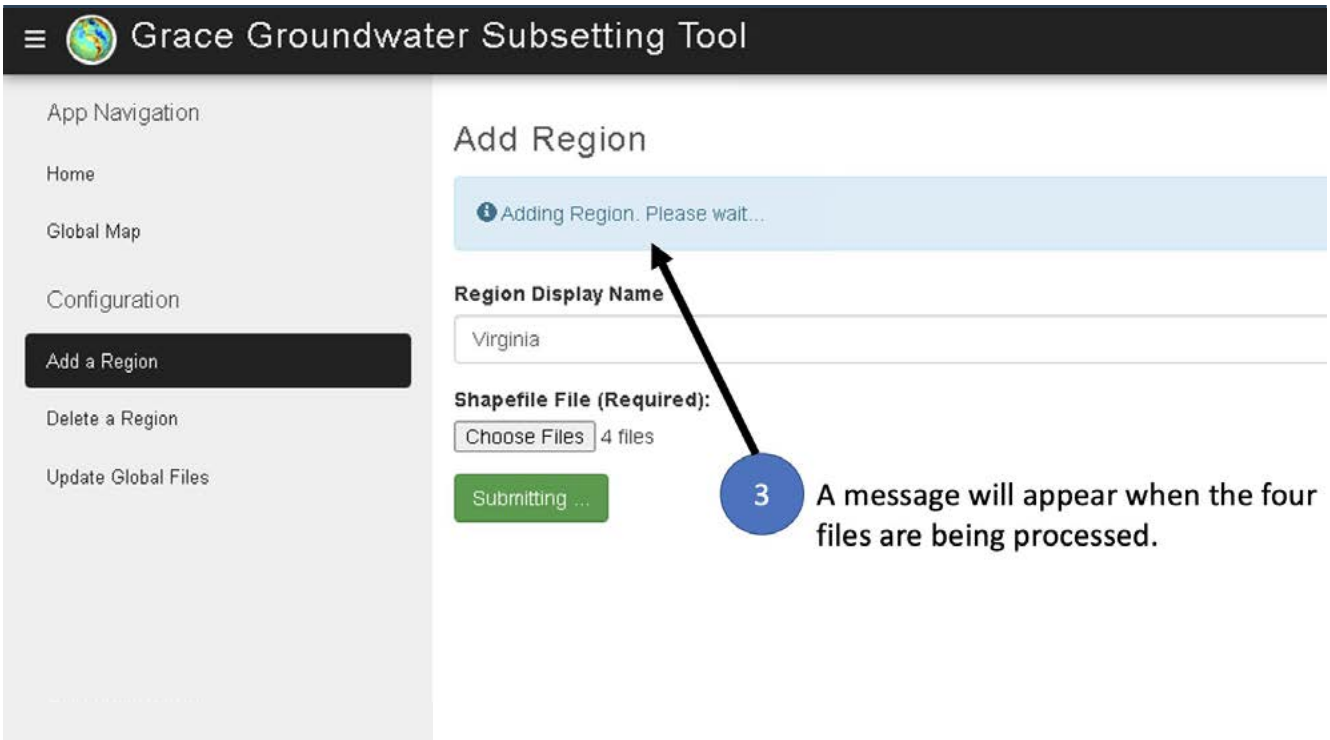


Figure 8. Adding a region in the GRACE Groundwater Subsetting Tool: step 4 screenshot

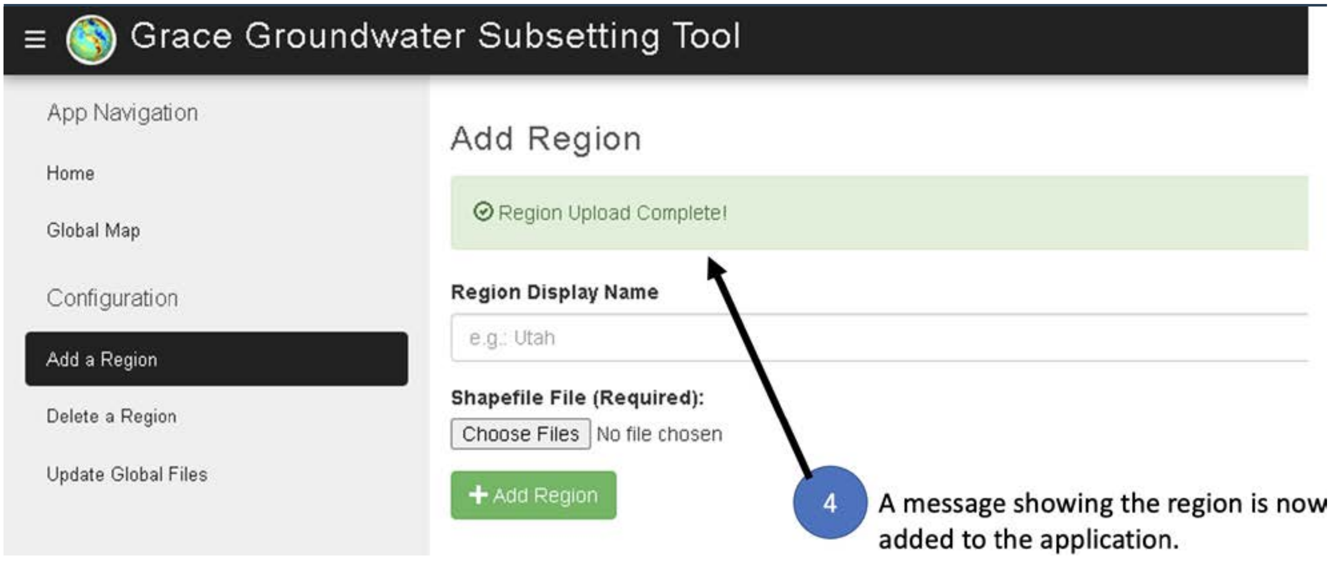
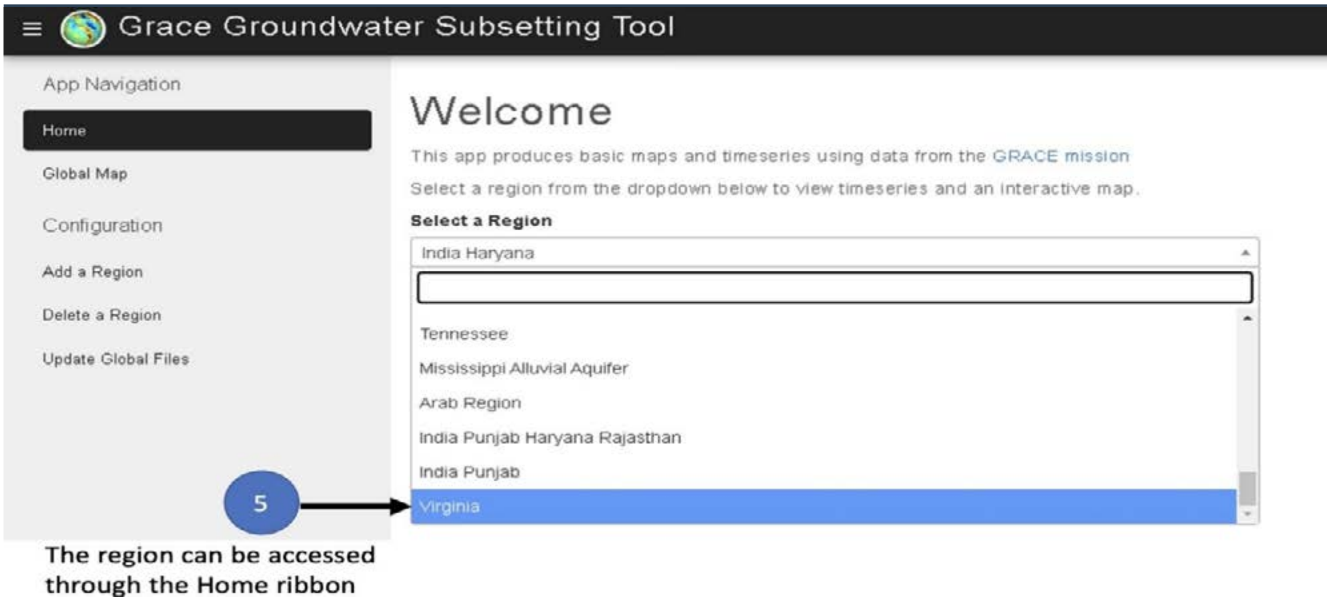
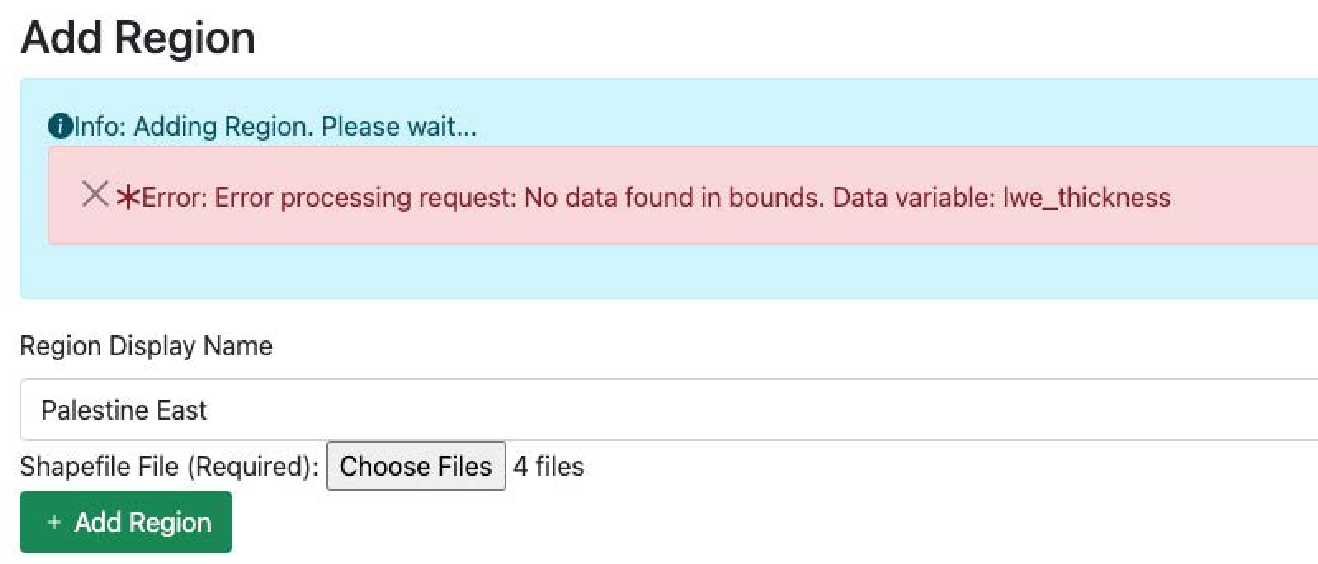


Figure 9. Adding a region in the GRACE Groundwater Subsetting Tool: step 5 screenshot



When uploading a region, it is recommended that the region be at least 3 x 3 degrees in size. Smaller regions can be processed, but uncertainty in the results increases. This is because the native GRACE grid cells are 3 x 3 degrees in resolution before downscaling to 0.5 x 0.5 degrees. GLDAS grid cells are 1 x 1 degree and therefore the resulting GWSa cells are at a 1 x 1 degree resolution. When uploading a region, the GGST algorithm searches to global GRACE and GLDAS grid cells to find cells where the centroid of the cells falls within the region shapefile. If the region is so small that no grid cells are found, the error message as shown in figure 10 is displayed.

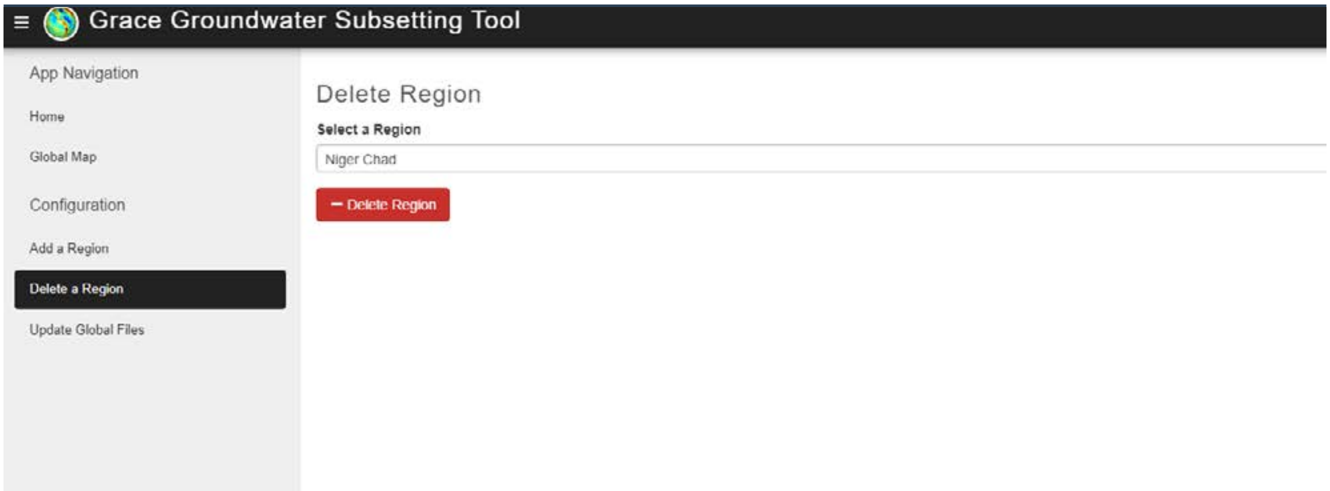
Figure 10. Add Region error message screenshot



In this case, the only solution is to perform a single point analysis with a latitude/longitude point within the region and GGST finds the single cell containing that point and returns the data sets (TWSa, GWSa, etc.) for that grid cell in time series format.

B. Deleting a region

Figure 11. Delete Region screenshot



Deleting a region is very simple. Proceed to the Delete a Region page. Select the region from the dropdown menu – see figure 11 – and hit the red "Delete Region" button. A message will display when the deletion has been completed.

4. Application programming interface

The Python API for GGST allows users to retrieve groundwater information about a point or region without having administrative privileges to the GGST web application. GGST API has four functions. Each of these functions requires different inputs and returns different results as desired by the user. The name of each function gives a glimpse of what each accomplishes. The four functions are:

- getStorageOptions.
- getPointValues.
- getRegionTimeseries.
- subsetRegionZipfile.

To run some of the functions listed above, the user will need an authentication token. Please refer to section 4.B on how to obtain a token. API can be implemented in many ways using a variety of coding languages and platforms. An example implementation using the Python code language in a Google Colaboratory notebook has been provided below. Before using API in the Google Colaboratory notebook, each of the four GGST API methods are explored in the following section.

A. Application programming interface methods

All four GSST API functions follow the same pattern as shown by the URL examples below. Each of the terms in brackets along with the parameters and values would be replaced by string values.

https://apps.geoglows.org/apps/[parent-app]/api/[MethodName]/?param1=value1&param2=value2&. . . paramN=valueN

To test API, the user will need a zip file of the region of interest. A set of sample files in the appropriate format has been provided. You may use your own files.

API_Fileset.zip (https://ggst.readthedocs.io/en/latest/_downloads/5e543765a38b66e62355c0a9fbd6d283/API_Fileset.zip)

Let's explore each API method individually and offer an example, see tables 1–4.

1. The getStorageOptions method

Table 1. get Storage Options

Parent application	GGST
Supported methods	GET
Returns	A JavaScript Object Notation (JSON) object with a list of storage options
Parameters	There are no parameters for the getStorageOptions function

Source: Authors.

Follow this link to inspect the JSON returned which lists the storage options available:

https://apps.geoglows.org/apps/ggst/api/getStorageOptions/

For simplicity, the options are given a variable name. For instance, the "Total Water Storage (GRACE)" has a variable name of "grace", and similarly the "Soil Moisture Storage (GLDAS)" is shortened to "sm".

2. The getPointValues method

Table 2. getPointValues

Parent application	GGST			
Supported methods	GET			
Returns	A JSON object with a time series for a given point			
Parameters	**Name**	**Description**	**Valid value**	**Required**
	Longitude	long in WGS 84 Proj	Any value on land with the GRACE Explorer Doman	Yes
			(-60,180)	
	Latitude	lat in WGS 84 Proj	Any value on land with the GRACE Explorer Doman (-60,90)	Yes
	storage_type	Storage type of interest	One of the abbreviated values from the first function.eg. grace, sw, sm or gw	Yes

Source: Authors.

Open the following example link to call API and inspect the JSON object returned (results will appear in a new window).

https://apps.geoglows.org/apps/ggst/api/getPointValues/?latitude=20.7&longitude=80.2&storage_type=gw

For the last two functions, the user will need to have an authentication token as it is required to run the code. It is best to call these two functions from Python. Please refer to the Google Colaboratory notebook for further instructions. See details in chapter 4.B on how to obtain one.

3. The getRegionTimeseries method

Table 3. getRegionTimeseries

Parent application	GGST			
Supported methods	POST			
Returns	A JSON object with area of the region, depletion time series, error range time series and storage time series			
Parameters	**Name**	**Description**	**Valid value**	**Required**
	Region name	Name for the subset region All files will have this name as prefix	String	Yes
	Storage type	storage type of interest	One of the abbreviated values from the first function, e.g., grace, sw, sm or gw	Yes
	files	A zipped folder	A zipped folder with .shp, .shx or .prj and .dbf files	Yes
	API token	Token from the Tethys portal	Token from a Tethys user account on the portal	Yes

Source: Authors.

Example query: files = {'shapefile': ("response.zip", uploaded["".join(uploaded)],'application/zip')} region_timeseries_request = requests.post("https://apps.geoglows.org/apps/ggst/api/getRegionTimeseries/", headers={"Authorization": f"Token {api_token}"}, data = {"name":"api_test", "storage_type": "tws"}, files=files) Response (trimmed for clarity): {'area': 437109427476.4769, 'depletion': [['2000-01-01', 0.0], ['2000-02-01', -273831.117], ['2000-03-01', -661208.652], …….. ['2021-09-01', 4792246.794]], 'error_range': [['2000-01-01', -6.045, -3.205], ['2000-02-01', -7.122, -3.798], ['2000-03-01', -8.648, -4.636], ['2021-09-01', 8.19, 11.796]], ……. 'success': 'success', 'values': [['2000-01-01', -4.625], ['2000-02-01', -5.46], …….. ['2021-09-01', 9.993]]}

4. The subsetRegionZipfile method

Here is an example query using the subsetRegionZipfile method.

Example Query: files = {'shapefile': ("response.zip", uploaded["".join(uploaded)],'application/zip')} subset_region_request = requests.post("https://apps.geoglows.org/apps/ggst/api/subsetRegionZipfile/", headers={"Authorization": f"Token {api_token}"}, data = {"name":"api_test"}, files=files) z = ZipFile(BytesIO(subset_region_request.content)) z.extractall()

The result will be a folder with numerical control files.

Table 4. subsetRegionZipfile

Parent application	GGST			
Supported methods	POST			
Returns	A zip file with regional netCDF files for each storage option clipped to the uploaded shapefile.			
Parameters	**Name**	**Description**	**Valid value**	**Required**
	Region name	Name for the subset region. All files will have this name as prefix	String	Yes
	Files	A zipped folder	A zipped folder with .shp .shx, .prj and .dbf files	Yes
	API token	Token from the Tethys portal	Token from a Tethys user account on the portal	Yes

Source: Authors.

B. Obtaining an authentication token

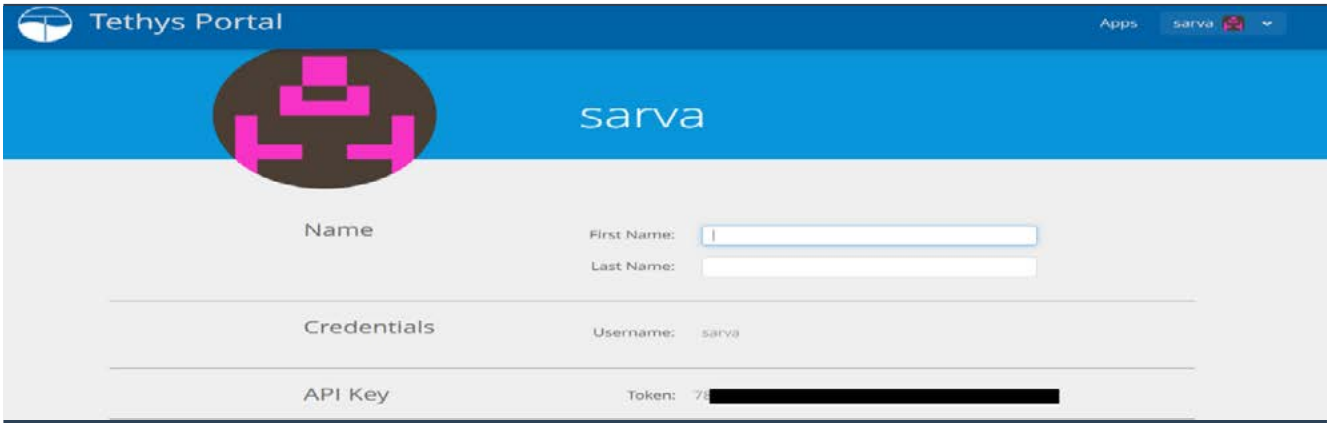
The last two functions of API require an authentication token. To obtain one, the user will need to sign up for an account on Geoglows Portal (https://apps.geoglows.org/) Click on the "Log In" button to get to the sign-up prompt.

Once signed in, click on your username in the upper right corner, opening a panel. Click on "User Settings" to reveal the API key, as shown in figure 12.

The authentication token or API key will be in the third section as shown in figure 13.

Figure 12. Retrieving the authentication token from Tethys portal: step 1 screenshot

Figure 13. Retrieving the authentication token from Tethys portal: step 2 screenshot



For privacy reasons, the remaining characters of this user's token have been hidden. It is also possible to request an authentication token directly from an administrator. It is recommended to use the sign-up method as it is faster.

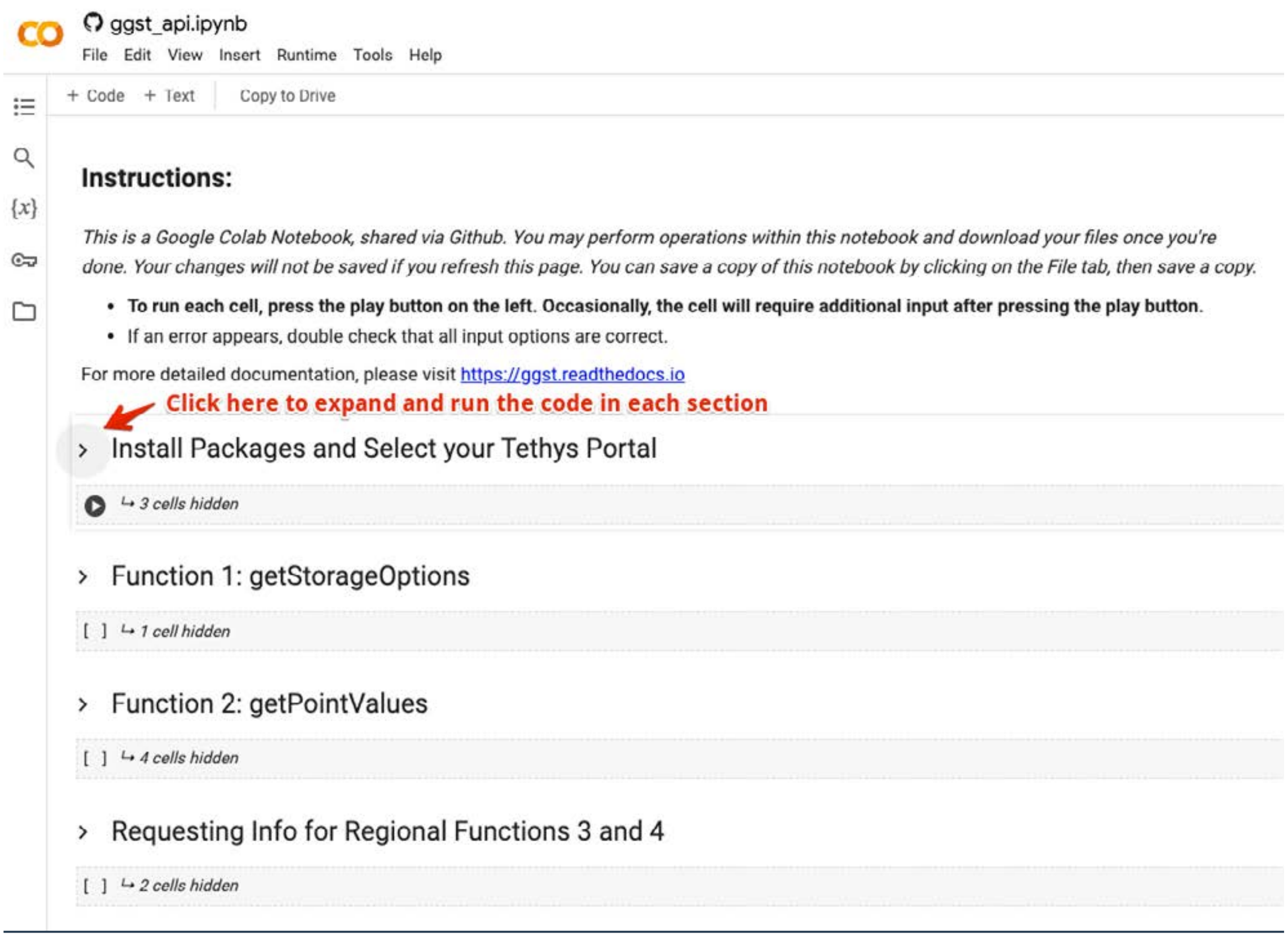
C. Groundwater Subsetting Tool application programming interface Google Colaboratory notebook

An example of calling GGST API using the Python coding language in a Google Colaboratory notebook has been provided which can be opened at this link:

https://colab.research.google.com/github/BYU-Hydroinformatics/ggst-notebooks/blob/main/ggst_api.ipynb

Run each cell of the notebook by hitting the play button on the left side of each cell and provide the necessary inputs by following the prompts. The notebook runs through all four of the API functions described in chapter 4, section A. To run some of the functions in this notebook, the user will have to sign up for a Tethys account and obtain an authentication token (API key) as explained in the previous sections.

Figure 14. Google Colaboratory notebook screenshot



The notebook is divided into multiple sections. Each section contains a set of cells, in each of which is a Python code. When you first launch the notebook, the sections are collapsed and you need to expand each section to view and run the code, see figure 14.

The cells should be run sequentially. To run a cell, click on run arrow in the upper left corner of the cell, as shown in figure 15.

Some cells require inputs as shown on the righthand side. You should enter the inputs before running the associated cells, see figure 16.

Some cells produce outputs when you run the cell. The outputs are displayed just below the cells, see figure 17.

The code is divided into six sections designed to help the user understand how to call each of the four functions and how to plot and visualize them.

Figure 15. Google Colaboratory notebook features screenshot

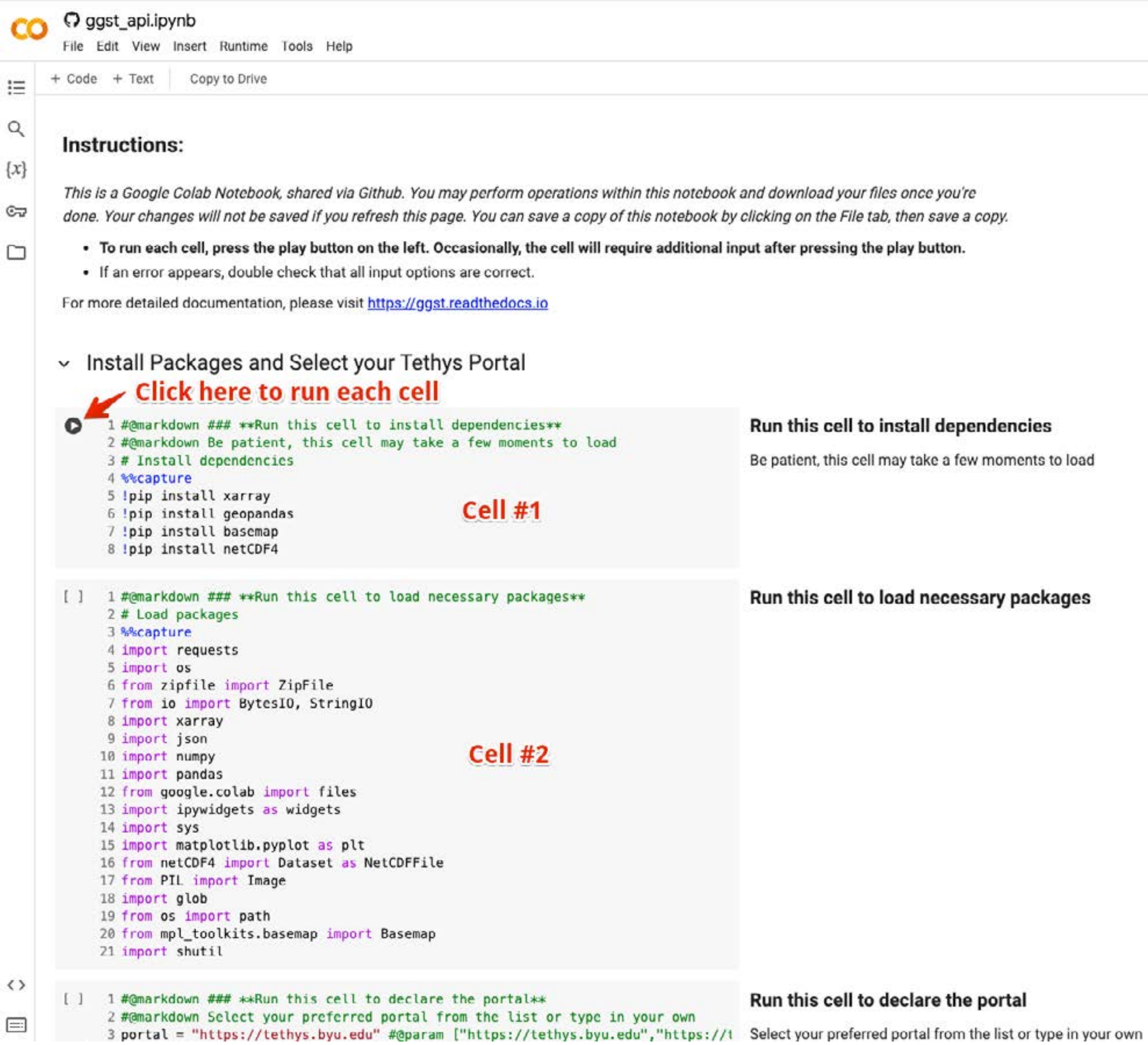
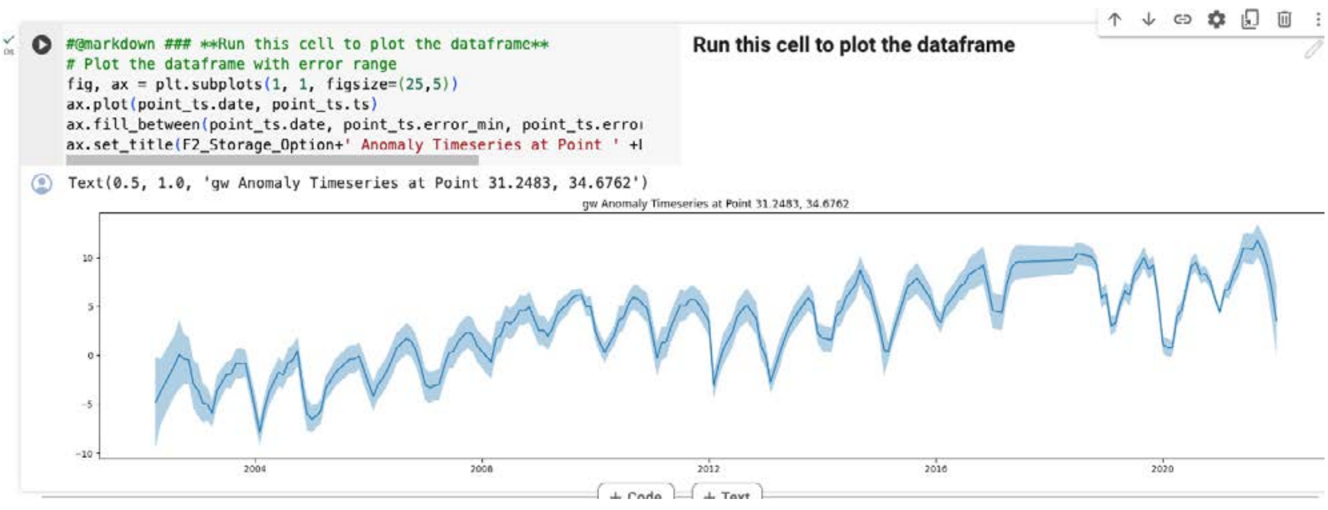


Figure 16. getPointValues input screenshot

Figure 17. getpointValues output screenshot



1. Install packages and select your Tethys portal:

In this section, dependencies and other python packages are installed and set up for the processing of the shapefile and rendering of the graph in latter cells. The dropdown menu lists all the available portals. A portal is a web hosting platform that executes the commands and returns the results as requested by the user. For this API, two portals are available: Tethys main and Geoglows.

2. Function 1: getStorageOptions

This cell lists all the available options and how to properly declare them in the appropriate cell.

3. Function 2: getPointValues

The user types in latitude and longitude coordinates and selects the desired storage option from a dropdown menu. The next several cells will create a dataframe, chart the time series and plot a graph with estimated error bars.

4. Requesting info for regional functions 3 and 4

The last two functions are regional functions and require more inputs to run. This section of the notebook walks the user through inputting that additional information. First, the user will be asked for their API token which must match their declared portal to work. Second, the user will be asked to give their region a name that will be used in naming the files. Lastly, the user will be asked to upload a zipped shapefile of the region of interest. This should contain four files (.shp, .shx, .prj and .dbf) zipped in a single folder.

5. Function 3: getRegionTimeseries

This asks for the user's desired storage option using a dropdown menu, calls the API, then displays an interactive table and graph of the data returned.

6. Function 4: getRegionZipfile

This calls API and returns a set of netCDF files which can be accessed from a tool bar on the left side of the screen as shown in figure 18.

Figure 18. getRegionZipfile output screenshot

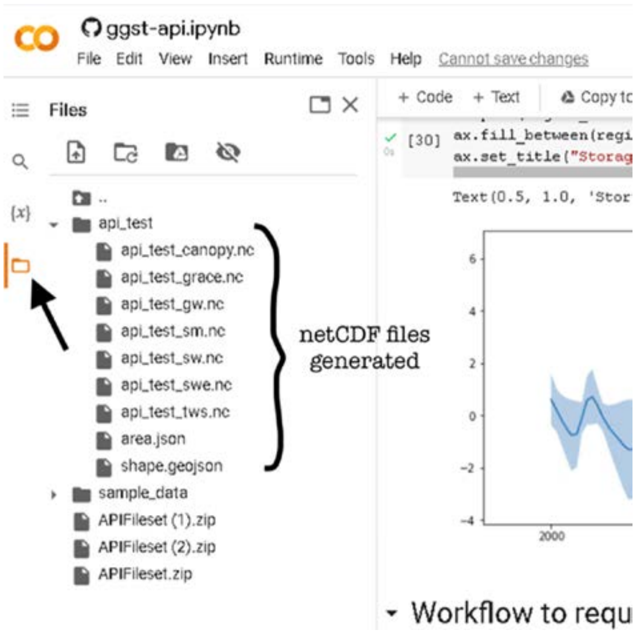


Table 5 elaborates on each of these files and their naming conventions.

This section will also help the user create a dataframe, plot their data and visualize the data on an animated map.

Table 5. Output files specifications

Name	Abbreviation	Source	Source Resolution
Total Water Storage	grace	GRACE	0.5 degrees
Surface Water Storage	sw	GLDAS	1.0 degrees
Soil Moisture Storage	sm	GLDAS	1.0 degrees
Groundwater Storage	gw	Calculated[a]	1.0 degrees
Snow Water Equivalent	swe	GLDAS	1.0 degrees
Terrestrial Water Storage	tws	GLDAS	1.0 degrees
Canopy Storage	canopy	GLDAS	1.0 degrees

[a] To learn more about how this is calculated please see chapter 2.

Source: Authors.

The water table fluctuation method

5

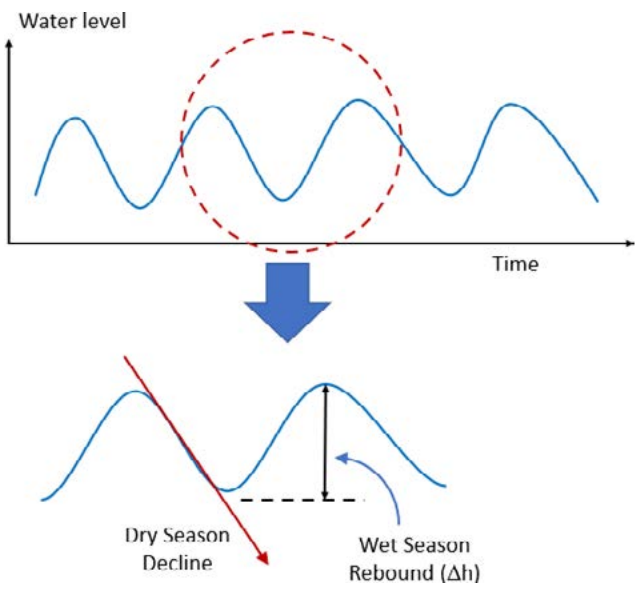
5. The water table fluctuation method

In addition to obtaining the GRACE-derived groundwater storage anomaly, it is possible to analyse the storage anomaly time series to extract an estimate of annual recharge using a technique called the water table fluctuation (WTF) method. The WTF method was originally developed to estimate recharge from seasonal fluctuations in groundwater levels measured directly in monitoring wells. When a water level time series exhibits seasonal fluctuations as shown in figure 19, it is assumed that the declining period during the dry part of the year results from pumping and groundwater discharge, and the rise during the wet part of the year is the result of recharge.

Using water levels derived from a monitoring well, the recharge can be estimated as:

where h is the rebound in water level, t is the time period (typically one year) and Sy is the specific yield or appropriate storage coefficient.

Figure 19. Water level fluctuations



Source: Authors.

The storage coefficient is necessary because the water level rise in the surrounding aquifer occurs in the fractional void space and the storage coefficient converts it to the appropriate liquid water equivalent component in the [length]/[time] infiltration rate units used by recharge. If this analysis is performed using the GWSa curve derived from GRACE, there is no need to use a storage coefficient as the anomaly is already in liquid water equivalent form and the recharge can be directly estimated as:

$$R = \frac{\Delta GWSa}{\Delta t}$$

where GWSa = the rise in groundwater extracted from the GRACE-derived GWSa curve.

$$R = S_y \frac{\Delta h}{t}$$

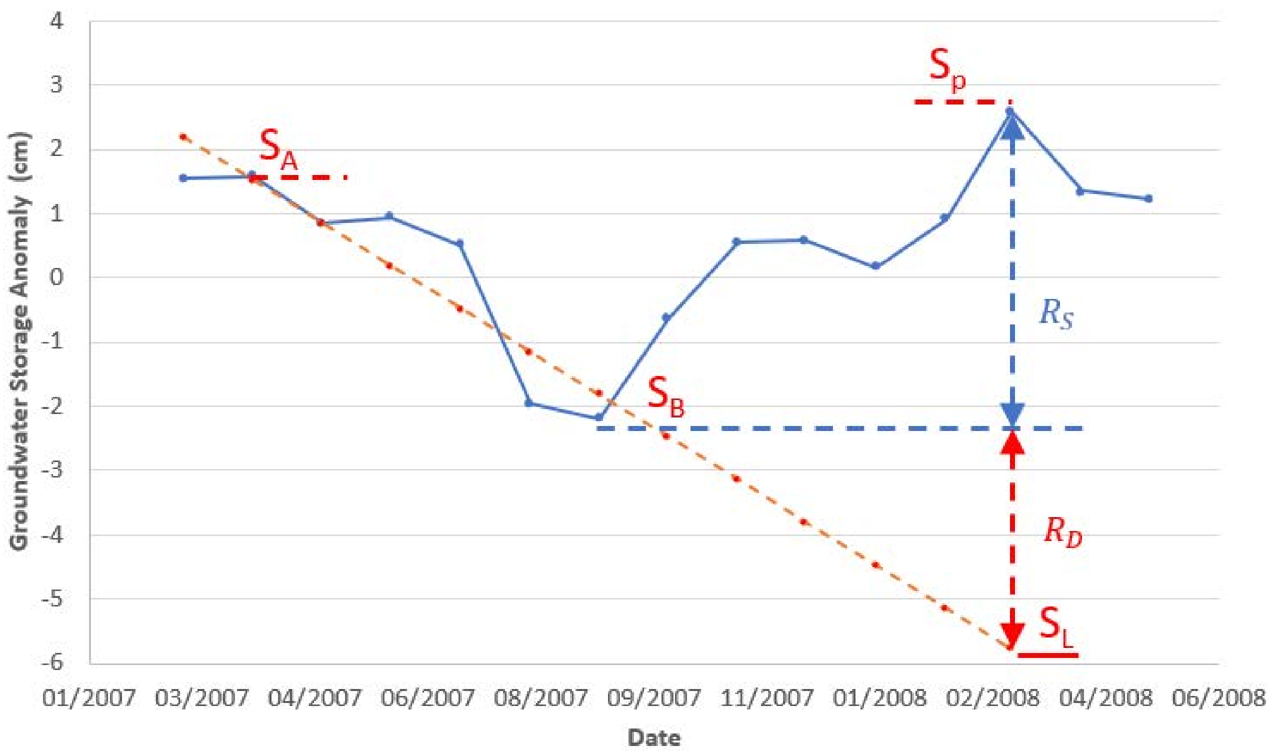
A. Methods for estimating recharge component

There are two general approaches for determining the height of the rise associated with recharge as shown in figure 20.

With the more conservative method, the rise is measured from the trough to the next peak as follows:

$$R_{method_1} = \frac{\Delta GWSa}{\Delta t} = \frac{S_p - S_B}{\Delta t} = R_S$$

Figure 20. Recharge calculation graph



Source: Authors.

Another method is to assume that the groundwater decline because of pumping and discharge continues at the same rate in the wet season and therefore the rise should be computed from a linear extrapolation of the declining line as follows:

$$R_{method_2} = \frac{\Delta GWSa}{\Delta t} = \frac{S_p - S_L}{\Delta t} = R_S + R_D$$

The recharge rates extracted from these two equations could be considered a low and a high estimate, although in the authors' experience the first method seems to be the most accurate. An example of applying the WTF method to estimate recharge in southern Niger can be found at: Evaluating Groundwater Storage Change and Recharge Using GRACE Data: A Case Study of Aquifers in Niger, West Africa (Barbosa et al., 2022)

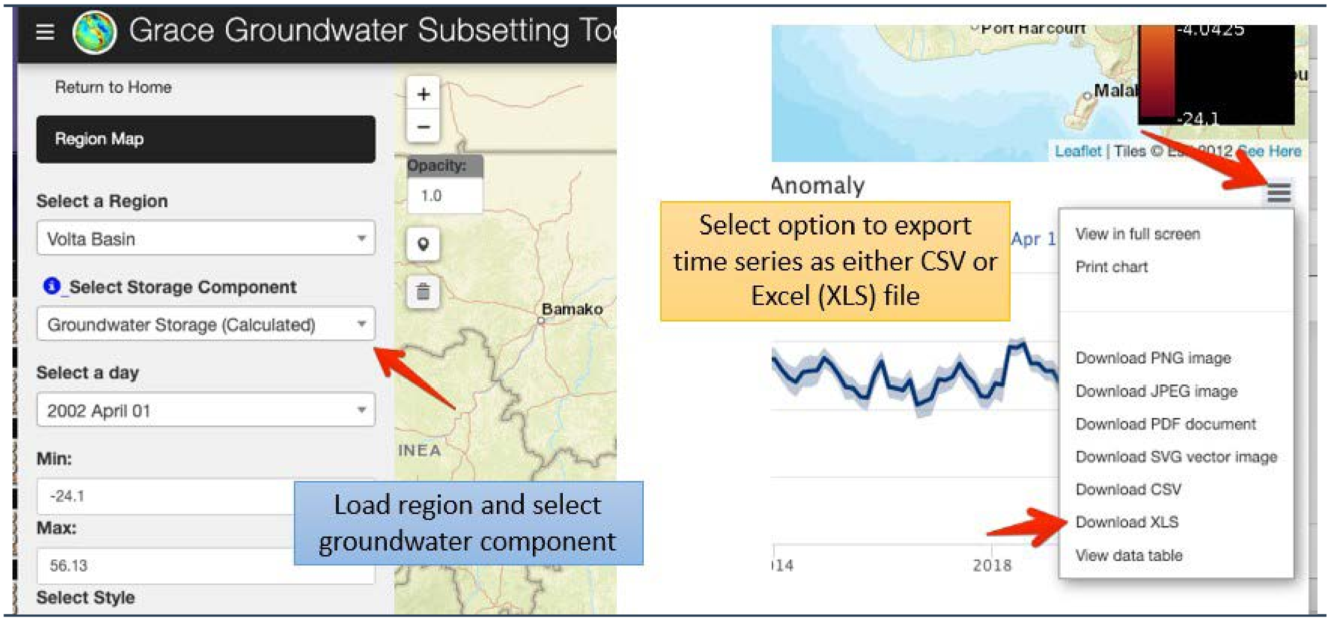
B. Downloading the water level time series from the Groundwater Subsetting Tool app

To apply the WTF method to estimate recharge on GRACE data, the user must first download the GWSa time series from the GGST app. To do so, first load the region and select Groundwater Storage (Calculated) under Select Storage Component, click on the three stacked lines in the upper right corner of the storage anomaly time series displayed and then download the time series as either a comma separated values (CSV) file or an Excel (XLS) file.

The storage anomaly chart is created, displayed and downloaded using the HighCharts plugin. The format of the resulting downloaded file is shown in figure 22.

The storage units are liquid water equivalent in cm, as expected, but the date units are reported in milliseconds since 1 January 1970. To convert to a more typical date unit, first create a new column and then enter the formula shown in figure 23

Figure 21. Downloading GWSa from the GGST app screenshot



for the first date in the list. This formula converts the number from milliseconds to days and then adds that number to the date value corresponding to 1 January 1970, thus creating a proper date value.

To see this value, change the number format to one of the standard date options. Whether it appears as month/day/year or day/month/year will depend on the user's regional settings.

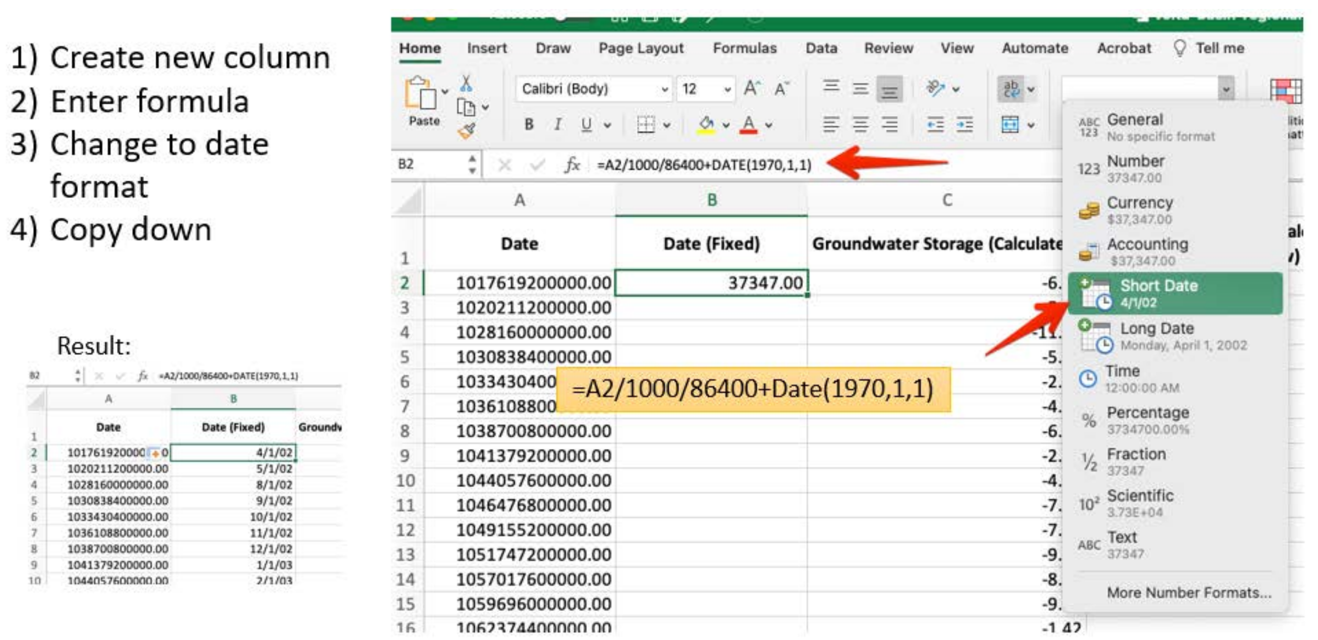
Figure 22. The downloaded GWSa Excel file screenshot

	A	B	C	D
1	Date	Groundwater Storage (Calculated)	Groundwater Storage (Calculated) Error Range (low)	Groundwater Storage (Calculated) Error Range (high)
2	1017619200000.00	-6.68	-18.99	5.64
3	1020211200000.00	-5.03	-11.75	1.68
4	1028160000000.00	-11.32	-15.81	-6.84
5	1030838400000.00	-5.27	-11.97	1.43
6	1033430400000.00	-2.46	-5.96	1.04
7	1036108800000.00	-4.25	-8.22	-0.27
8	1038700800000.00	-6.55	-10.69	-2.42
9	1041379200000.00	-2.35	-7.66	2.96
10	1044057600000.00	-4.07	-8.03	-0.11
11	1046476800000.00	-7.17	-10.09	-4.25
12	1049155200000.00	-7.65	-10.02	-5.27
13	1051747200000.00	-9.07	-11.38	-6.77
14	1057017600000.00	-8.68	-10.87	-6.49
15	1059696000000.00	-9.00	-10.78	-7.23
16	1062374400000.00	-1.42	-4.87	2.04

cm — cm — cm

milliseconds since january 1, 1970

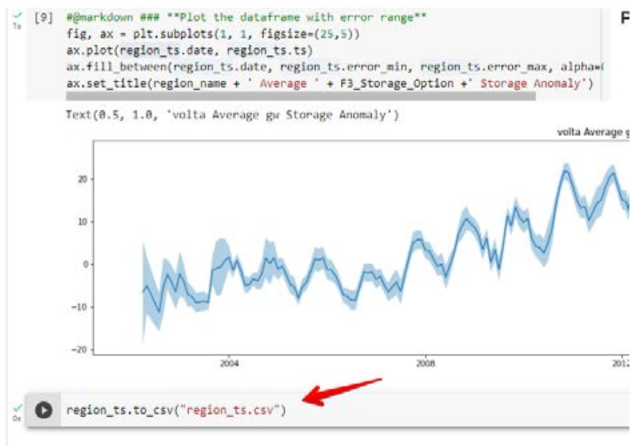
Figure 23. Steps for fixing the date format screenshot



C. Downloading the water level time series from the application programming interface Google Colaboratory notebook

The time series can be downloaded directly from the sample Google Colaboratory API Python script. After uploading a region shapefile and then generating and plotting the storage anomaly time series, run the line of code to export the Python Pandas dataframe containing the time series to a CSV file, see figure 24.

Figure 24. Exporting the output time series to a CSV file: step 1



This file will then appear in the files section of the Google Colaboratory notebook interface on the left. Click the three vertical dots to the right of the file and select the Download option, see figure 25.

In this case, the resulting CSV file has the dates in the correct format and no changes are necessary, as shown in figure 26.

Figure 25. Exporting the output time series to a CSV file: step 2

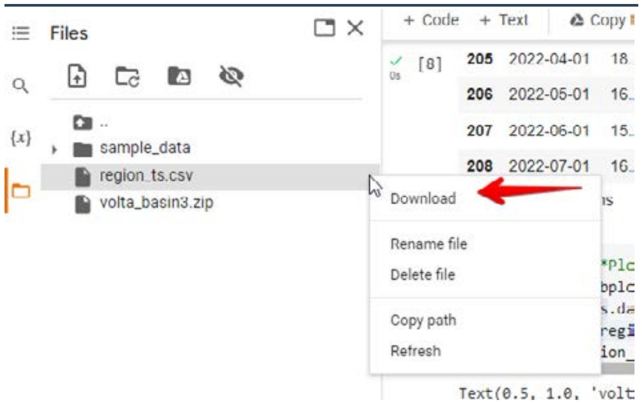


Figure 26. The output CSV file

	A	B	C	D	E	F
1		date	ts	error_min	error_max	
2	0	4/1/2002	-6.66	-18.978	5.658	
3	1	5/1/2002	-5.033	-11.749	1.683	
4	2	8/1/2002	-11.323	-15.81	-6.835	
5	3	9/1/2002	-5.273	-11.971	1.426	
6	4	10/1/2002	-2.464	-5.961	1.034	
7	5	11/1/2002	-4.24	-8.214	-0.267	
8	6	12/1/2002	-6.551	-10.684	-2.419	
9	7	1/1/2003	-2.345	-7.653	2.962	
10	8	2/1/2003	-4.075	-8.033	-0.118	
11	9	3/1/2003	-7.156	-10.076	-4.237	
12	10	4/1/2003	-7.638	-10.015	-5.261	
13	11	5/1/2003	-9.063	-11.364	-6.762	
14	12	7/1/2003	-8.686	-10.879	-6.494	
15	13	8/1/2003	-9.002	-10.775	-7.229	
16	14	9/1/2003	-1.429	-4.881	2.022	
17	15	10/1/2003	-0.977	-7.424	5.47	
18	16	11/1/2003	-0.676	-7.373	6.021	

D. Gaps in the gravity recovery and climate experiment data

If the user carefully inspects the groundwater storage time series CSV file, the user will see that there are several missing months or gaps in the data. For example, in figure 27, the month of June is missing in 2003.

Figure 27. Screenshot showing the missing months

	A	B	C	D	E	F
1		date	ts	error_min	error_max	
2	0	4/1/2002	-6.66	-18.978	5.658	
3	1	5/1/2002	-5.033	-11.749	1.683	
4	2	8/1/2002	-11.323	-15.81	-6.835	
5	3	9/1/2002	-5.273	-11.971	1.426	
6	4	10/1/2002	-2.464	-5.961	1.034	
7	5	11/1/2002	-4.24	-8.214	-0.267	
8	6	12/1/2002	-6.551	-10.684	-2.419	
9	7	1/1/2003	-2.345	-7.653	2.962	
10	8	2/1/2003	-4.075	-8.033	-0.118	
11	9	3/1/2003	-7.156	-10.076	-4.237	
12	10	4/1/2003	-7.638	-10.015	-5.261	
13	11	5/1/2003	[illegible].063	-11.364	-6.762	
14	12	7/1/2003	[illegible].686	-10.879	-6.494	
15	13	8/1/2003	-9.002	-10.775	-7.229	
16	14	9/1/2003	-1.429	-4.881	2.022	
17	15	10/1/2003	-0.977	-7.424	5.47	
18	16	11/1/2003	-0.676	-7.373	6.021	
19	17	12/1/2003	0.771	-4.438	5.979	

This is because there were periods when the GRACE satellites did not produce usable data. The largest gap is a 12-month period in 2017–2018 between the end of the original GRACE mission in 2017 and when the subsequent GRACE-FO satellites were launched and became operational in 2018. Figure 28 shows a sample plot for an aquifer in southern Niger with the gaps shown.

For the years with large gaps, it can be difficult to identify seasonal trends and apply the WTF method. One way to resolve this problem is to use a statistical algorithm to detect seasonal patterns in the data and impute synthetic data in the gaps. This can be accomplished using a simple seasonal decomposition model (statsmodels.tsa.seasonal.seasonal_decompose) implemented in the statsmodels Python package to impute the missing data. This model first removes the trend using a convolution filter (the trend component), then computes the average value for each period (the seasonal component), in this case months, with the residual component being the difference between the monthly average (seasonal component) and the actual monthly measurements. With this approach, the GWSa time series is decomposed into three components: the trend, the seasonal and the random components:

$$Y\ [t] = T\ [t] + S[t] + e[t]$$

where Y[t] is GWSa, T[t] is the GWSa trend, S[t] is the seasonal GWSa component and e[t] is the residual GWSa component. The decomposition components for the data shown above are as illustrated in figure 29.

Figure 28. Sample GWSa plot for an aquifer in southern Niger with data gaps

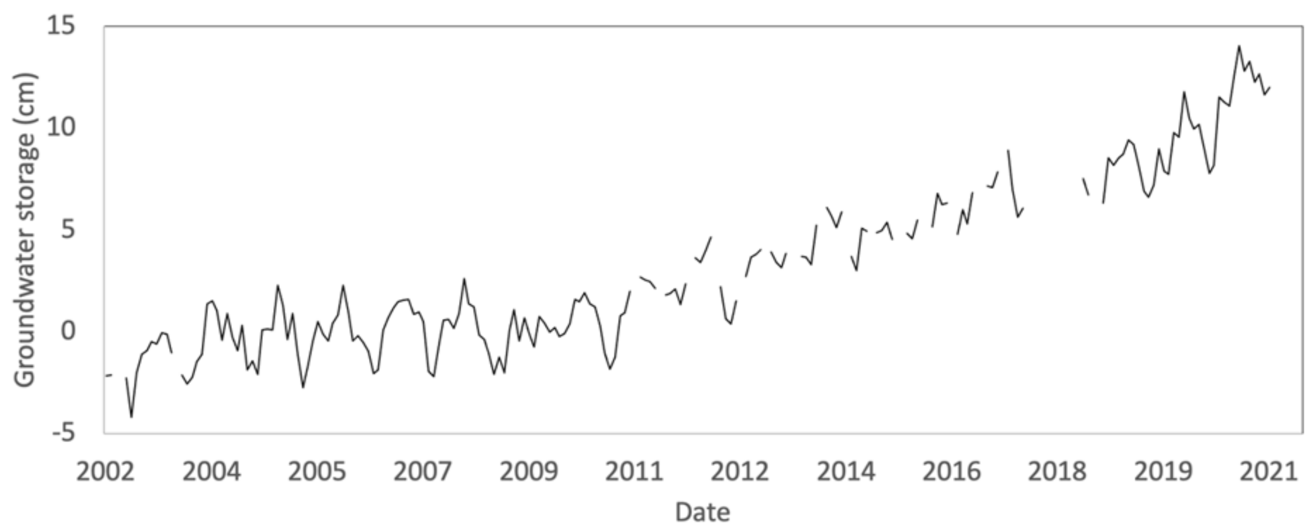


Figure 29. Decomposition components for the GWSa time series data

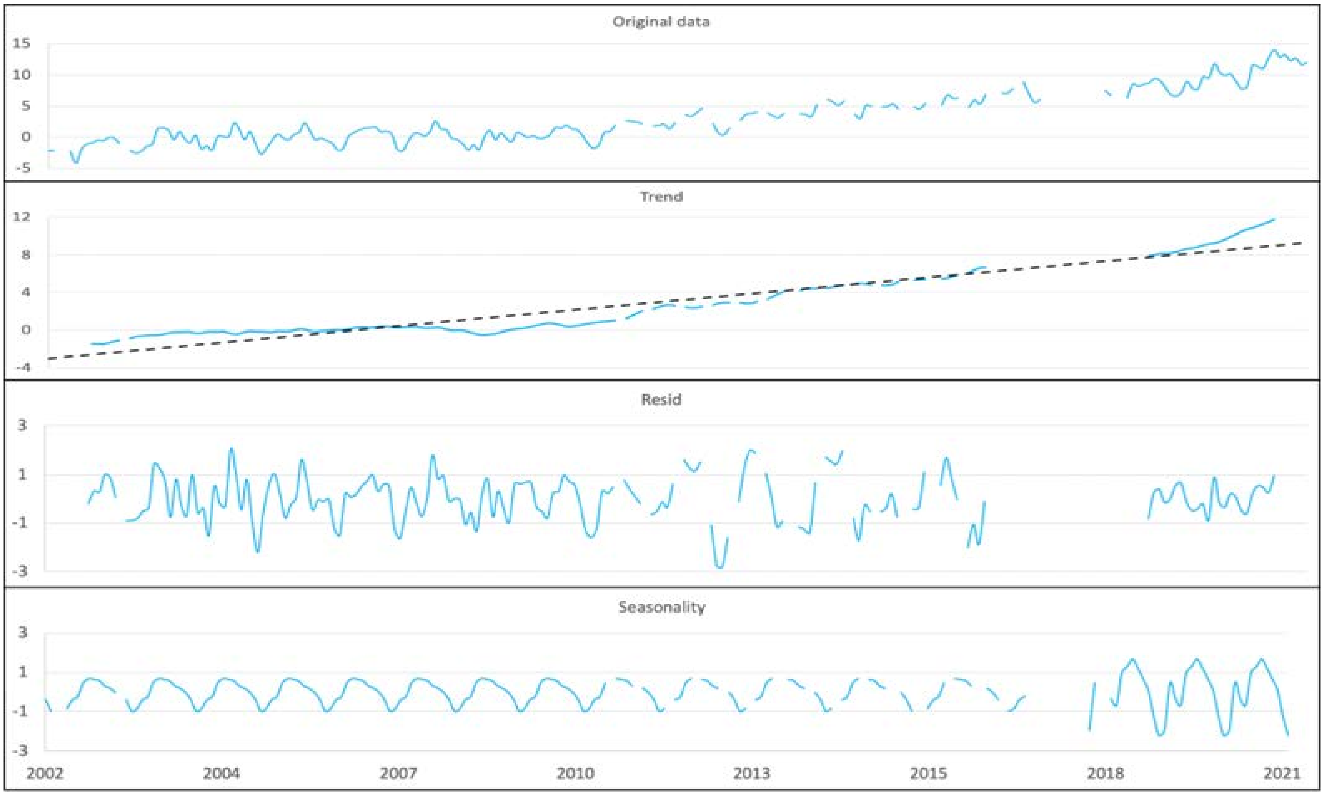
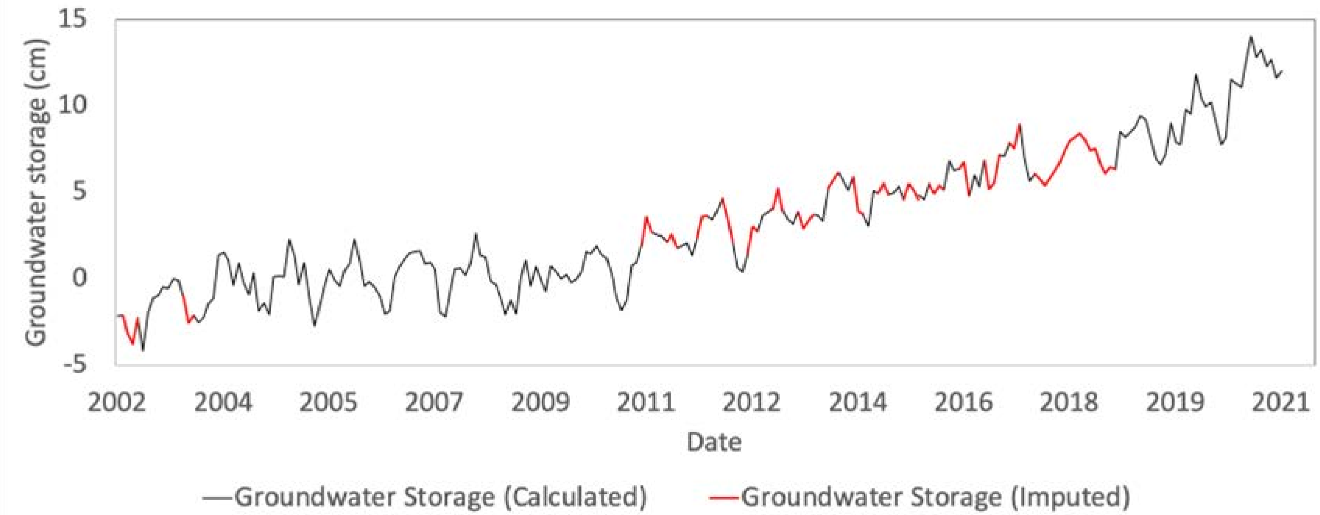


Figure 30. GWSa timeseries with imputed values in red



To impute the missing data, use the trend from the data decomposition then add the average of the monthly and residual values for that month to estimate the missing value. This model can be written as:

$$Y\ [t] = y(T\ \ [t]) + S[t] + e[t]$$

Figure 30 shows the original time series in black, with imputed values in red.

E. Data imputation tools

To assist users in applying the statsmodel method to impute gaps in the GRACE data, we have implemented a Python code to perform the imputation in a Google Colabatory notebook – see link.

https://colab.research.google.com/github/BYU-Hydroinformatics/ggst-notebooks/blob/main/impute_gaps_GRACE.ipynb

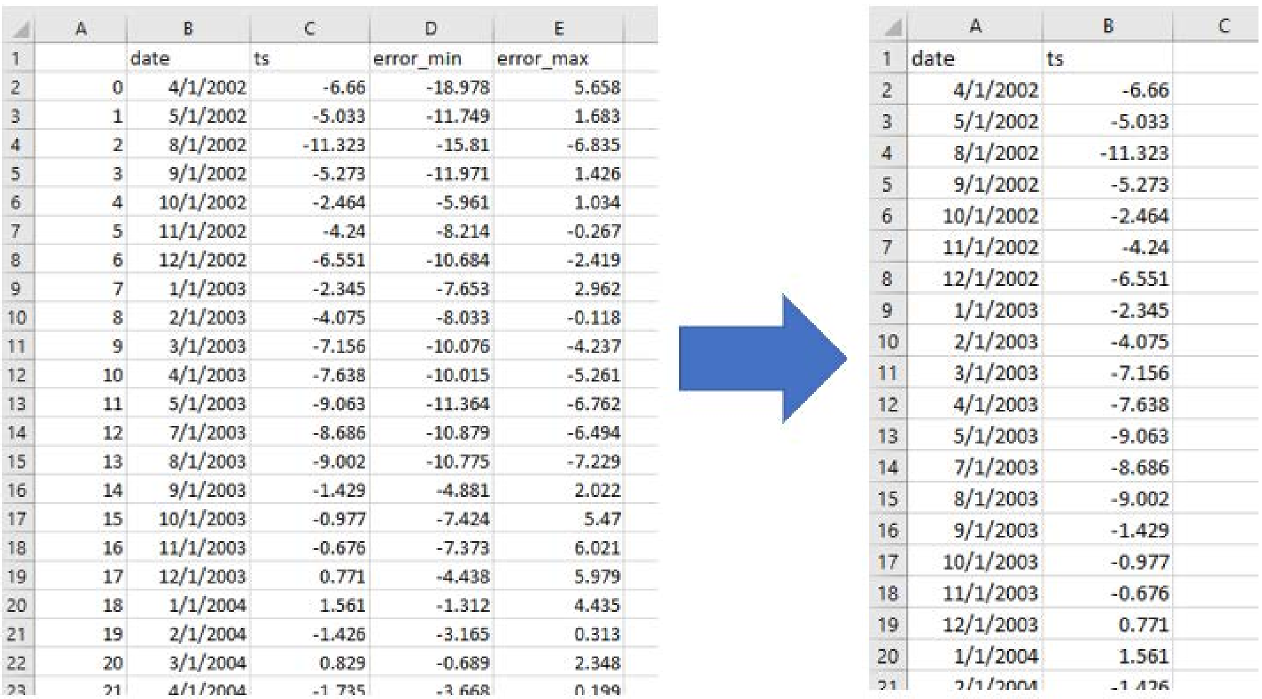
After launching the notebook, follow the instructions in the code. Before running the code, the user will need to prepare and upload a CSV file with the original data with the gaps. This file will need to contain only two columns which the user can copy and paste from the full CSV and then save as a separate CSV file ("base_file.csv" for example), as shown in figure 31.

At this point, the file is ready to be used with the Google Colabatory notebook.

Here is a sample file you can use with the script: west-gwsa-raw-clean.csv

https://ggst.readthedocs.io/en/latest/_downloads/2a3c7527df3de835c68ed6347fa4496f/west-gwsa-raw-clean.csv.

Figure 31. Preparation of the CSV file to be uploaded screenshot



	A	B	C	D	E
1		date	ts	error_min	error_max
2	0	4/1/2002	-6.66	-18.978	5.658
3	1	5/1/2002	-5.033	-11.749	1.683
4	2	8/1/2002	-11.323	-15.81	-6.835
5	3	9/1/2002	-5.273	-11.971	1.426
6	4	10/1/2002	-2.464	-5.961	1.034
7	5	11/1/2002	-4.24	-8.214	-0.267
8	6	12/1/2002	-6.551	-10.684	-2.419
9	7	1/1/2003	-2.345	-7.653	2.962
10	8	2/1/2003	-4.075	-8.033	-0.118
11	9	3/1/2003	-7.156	-10.076	-4.237
12	10	4/1/2003	-7.638	-10.015	-5.261
13	11	5/1/2003	-9.063	-11.364	-6.762
14	12	7/1/2003	-8.686	-10.879	-6.494
15	13	8/1/2003	-9.002	-10.775	-7.229
16	14	9/1/2003	-1.429	-4.881	2.022
17	15	10/1/2003	-0.977	-7.424	5.47
18	16	11/1/2003	-0.676	-7.373	6.021
19	17	12/1/2003	0.771	-4.438	5.979
20	18	1/1/2004	1.561	-1.312	4.435
21	19	2/1/2004	-1.426	-3.165	0.313
22	20	3/1/2004	0.829	-0.689	2.348
23	21	4/1/2004	-1.735	-3.668	0.199

	A	B	C
1	date	ts	
2	4/1/2002	-6.66	
3	5/1/2002	-5.033	
4	8/1/2002	-11.323	
5	9/1/2002	-5.273	
6	10/1/2002	-2.464	
7	11/1/2002	-4.24	
8	12/1/2002	-6.551	
9	1/1/2003	-2.345	
10	2/1/2003	-4.075	
11	3/1/2003	-7.156	
12	4/1/2003	-7.638	
13	5/1/2003	-9.063	
14	7/1/2003	-8.686	
15	8/1/2003	-9.002	
16	9/1/2003	-1.429	
17	10/1/2003	-0.977	
18	11/1/2003	-0.676	
19	12/1/2003	0.771	
20	1/1/2004	1.561	
21	2/1/2004	-1.426	

F. Multi-linear trend analysis

In the seasonal decomposition method for gap imputation, a single linear trend was described. Figure 32 shows the trend resulting from the sample file linked above with a single trend line:

Figure 32. Trend resulting from the GWSa sample file with a single trend line

However, many data sets exhibit multiple linear trends. For this data set, there are four distinct trends. The Python script has an option to perform a multi-linear regression analysis. For this data set, we set the number_breakpoints variable to 3, and run a multi-linear regression algorithm that fits the data as shown in figure 33.

Note that three interior breakpoints result in four linear trends. This option results in the trends as shown in figure 34.

And finally, the gap imputation with four trend lines results is shown in figure 35.

Figure 33. Multi-linear regression analysis result

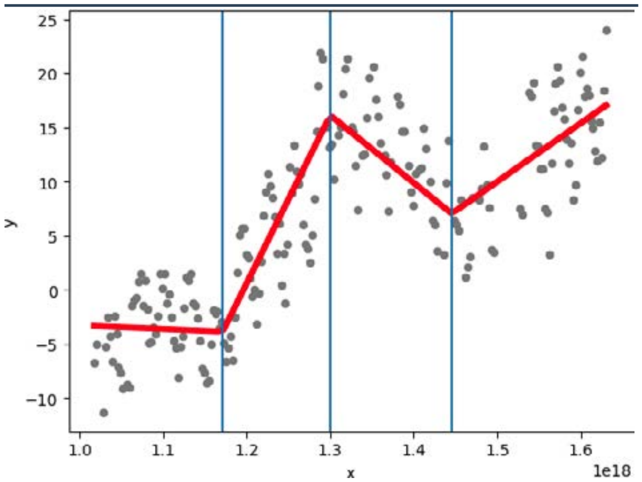


Figure 34. Four linear trends result

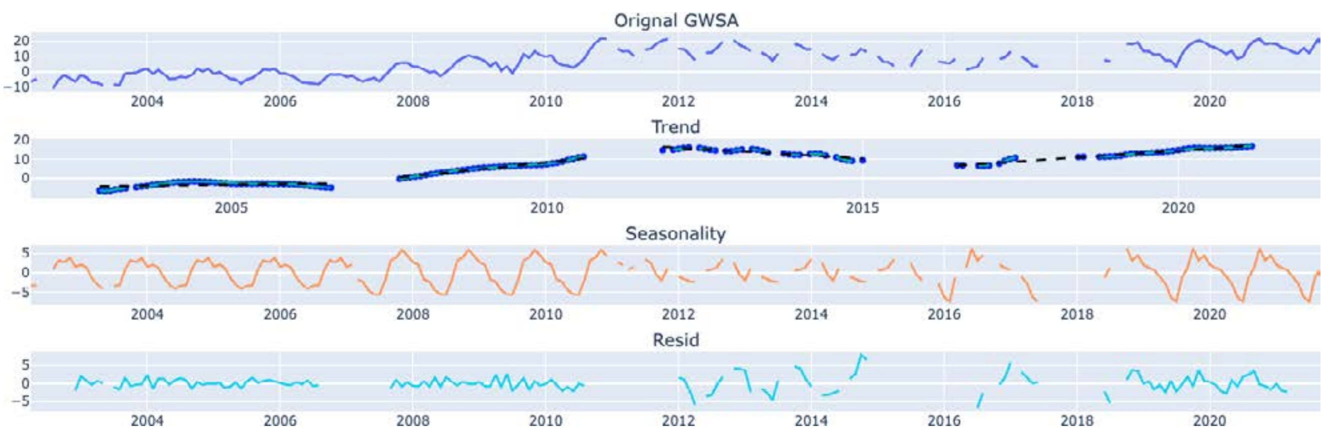
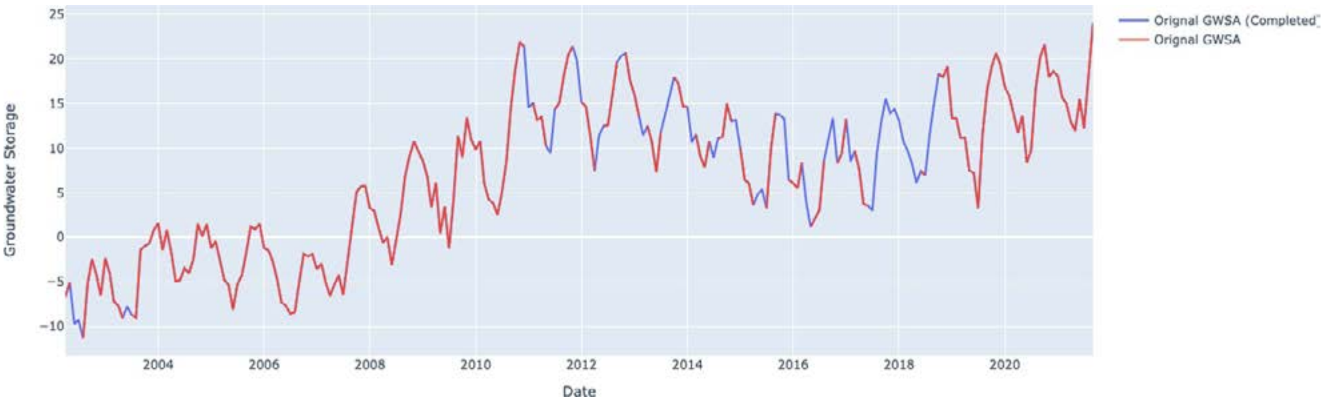


Figure 35. The gap imputation result



G. Data processing examples

Once the gaps have been filled, the last step is to plot and analyse the curves one season at a time, extract the GWSa values from the curve, and calculate the recharge estimate using either method 1 or method 2, see figure 36.

The following Excel file illustrates how to examine and process each season of data from a GRACE-derived and imputed GWSa time series: west-gwsa-wtf.xlsx.

https://ggst.readthedocs.io/en/latest/_downloads/8e545f1f86a93a2d93c5fbb0e028c03a/west-gwsa-wtf.xlsx

After opening the file, copy-paste the GWSa values generated by the imputation algorithm as shown in figure 37. Note that the imputed values have more digits than the original values. The formulas in columns C and D separate the imputed data in column B to allow a multi-coloured plot where the original imputed sections can be clearly visualized in orange.

Figure 36. Calculation of the recharge estimate after gap imputation: step 1

	A	B	C	D	E	F	G
1	Year:	2014					
2							
3	Date	GWSa	Orig	Imputed			
4	2014-04-01	4.14	4.14	#N/A			
5	2014-05-01	4.49	4.49	#N/A		SP	8
6	2014-06-01	5.89	5.89	5.89		SB	0.2
7	2014-07-01	7.24	#N/A	7.24		SL	-8
8	2014-08-01	7.20	7.20	7.20			
9	2014-09-01	8.74	8.74	#N/A		RS	7.8
10	2014-10-01	7.50	7.50	#N/A		RD	8.2
11	2014-11-01	6.83	6.83	6.83			
12	2014-12-01	2.91	#N/A	2.91		R1	7.8
13	2015-01-01	3.19	3.19	3.19		R2	16
14	2015-02-01	0.54	0.54	#N/A			
15	2015-03-01	0.38	0.38	#N/A			
16	2015-04-01	2.38	2.38	2.38			
17	2015-05-01	4.28	#N/A	4.28			
18	2015-06-01	5.99	#N/A	5.99			
19	2015-07-01	6.98	6.98	6.98			
20	2015-08-01	7.39	7.39	#N/A			
21	2015-09-01	7.93	7.93	7.93			
22	2015-10-01	7.65	#N/A	7.65			
23	2015-11-01	6.62	#N/A	6.62			
24							
25							

Figure 37. Calculation of the recharge estimate after gap imputation: step 2

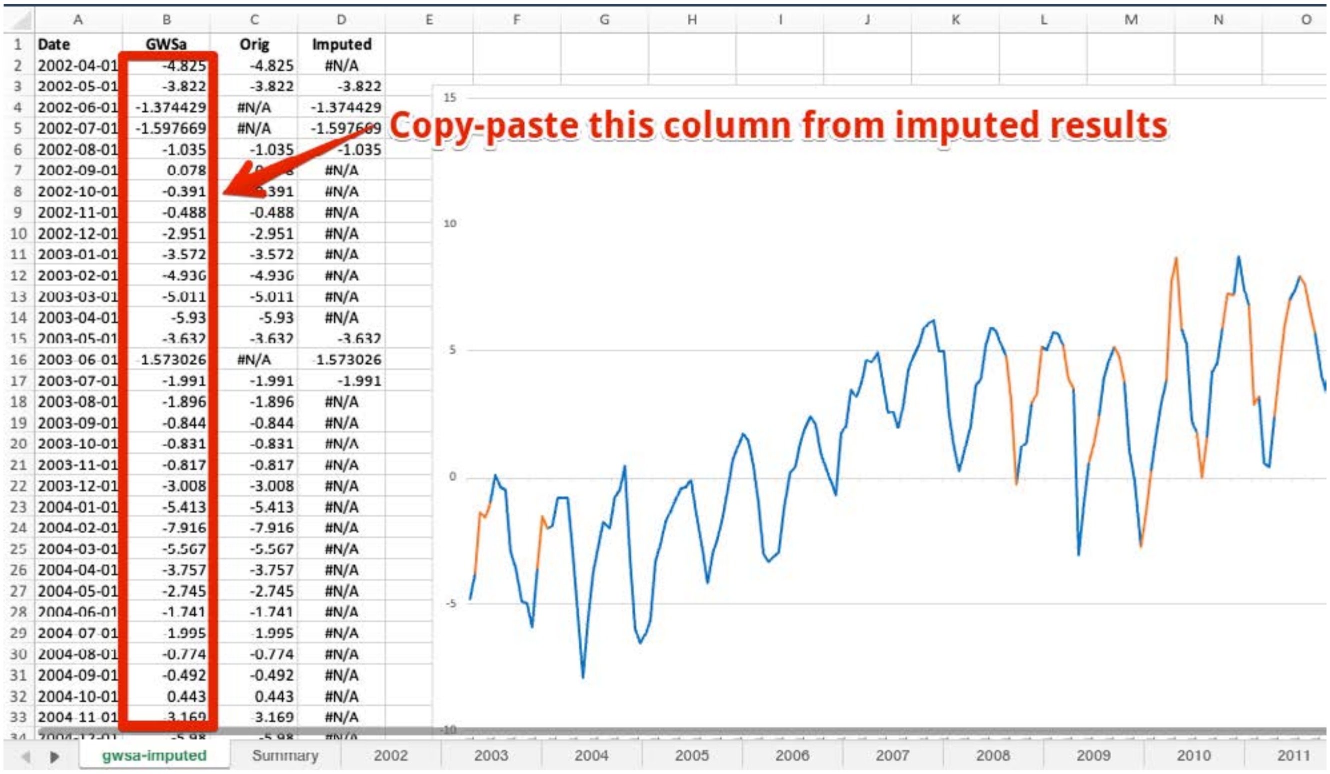
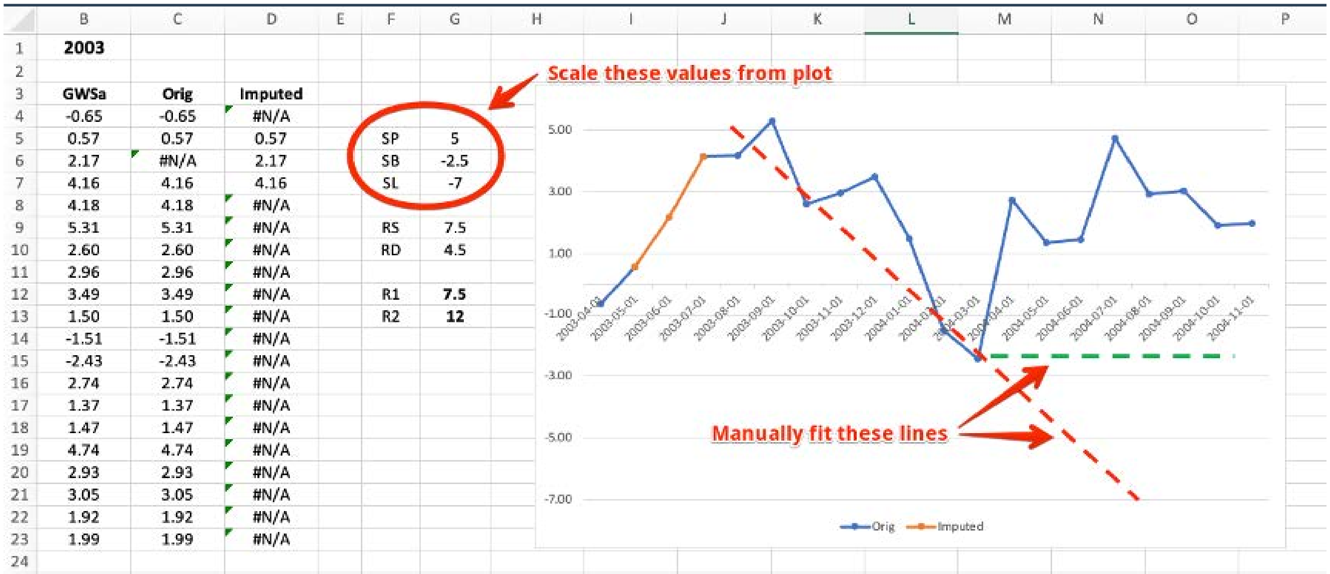


Figure 38. Calculation of the recharge estimate after gap imputation: step 3

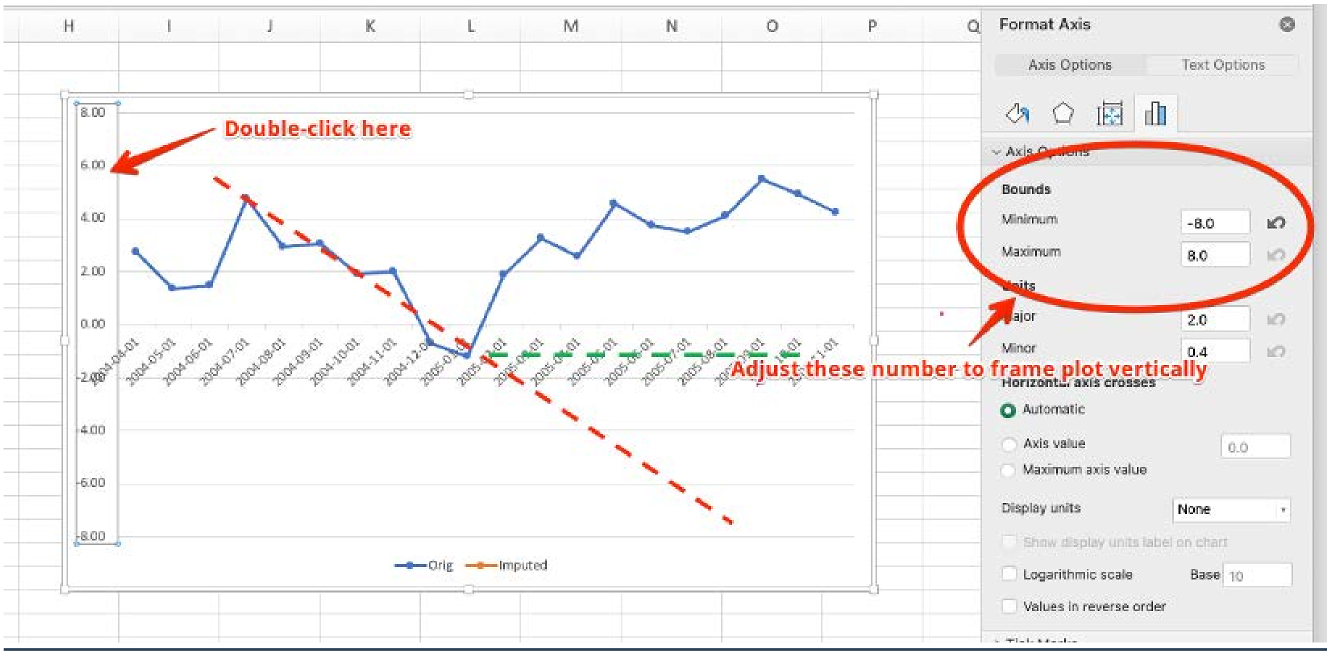


At this point the user can browse through each of the tabs for the years starting in 2002. On each page, the seasonal values are automatically pulled from the main sheet using a VLOOKUP formula. For each page, manually adjust the red and green lines to fit the descending branch and the base, see figure 38. Then manually scale off the SP, SB and SL values in cm from the vertical axis and enter into the three cells indicated in the diagram. The RS, RD, R1 and R2 values will then be automatically calculated.

As you examine the plot for each year, you may need to adjust the range of the vertical axis before you can properly fit the lines. To do this, double-click on the vertical axis, click on the axis options tab and manually adjust the minimum and maximum bounds to properly frame the plot, see figure 39.

If you need to add additional years, copy one of the yearly sheets, rename it and change the year at the top of the sheet. After processing all the years and calculating all of the R1 and R2 values, you can see a summary in the Recharge Rate Summary sheet, see figure 40.

Figure 39. Calculation of the recharge estimate after gap imputation: step 4



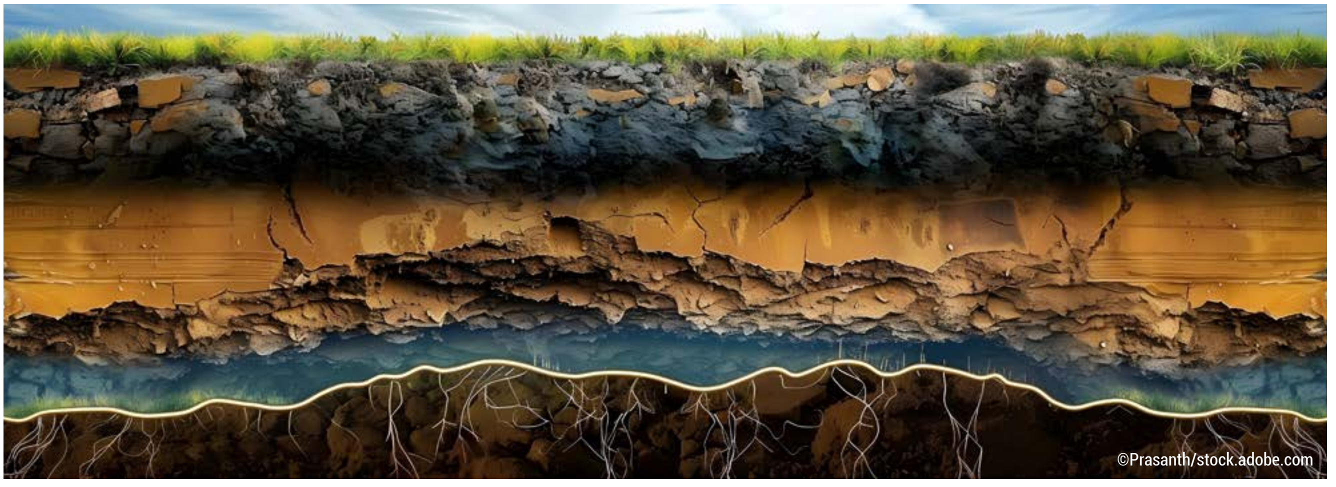
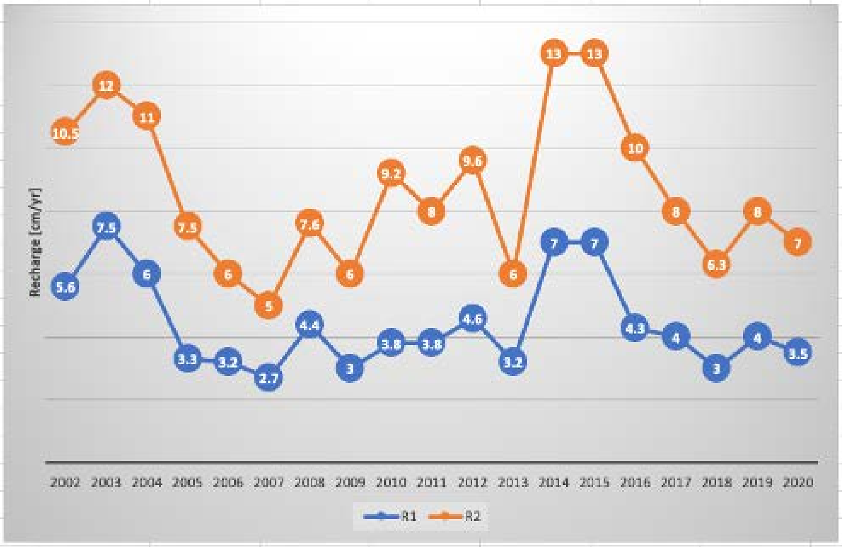


Figure 40. Calculation of the recharge estimate after gap imputation: output

Recharge Rate Sumary

Year	R1	R2
2002	5.6	10.5
2003	7.5	12
2004	6	11
2005	3.3	7.5
2006	3.2	6
2007	2.7	5
2008	4.4	7.6
2009	3	6
2010	3.8	9.2
2011	3.8	8
2012	4.6	9.6
2013	3.2	6
2014	7	13
2015	7	13
2016	4.3	10
2017	4	8
2018	3	6.3
2019	4	8
2020	3.5	7
Mean	4.4	8.6



References

Barbosa, S.A. and others, 2022. Evaluating Groundwater Storage Change and Recharge Using GRACE Data: A Case Study of Aquifers in Niger, West Africa. Remote Sensing 14, 1532. https://doi.org/10.3390/rs14071532 available at https://www.mdpi.com/2072-4292/14/7/1532.

Famiglietti, J.S. and others, 2011. Satellites measure recent rates of groundwater depletion in California's Central Valley. Geophysical Research Letters 38. https://doi.org/10.1029/2010GL046442 available at https://onlinelibrary.wiley.com/doi/abs/10.1029/2010GL046442.

Jones, N. and others, 2014. Tethys: A Software Framework for Web-Based Modeling and Decision Support Applications, in: Ames, D.P., Quinn, N.W.T., Rizzoli, A.E. (Eds.), . Presented at the Proceedings of the 7th International Congress on Environmental Modelling and Software, International Environmental Modelling and Software Society (iEMSs). Available at http://www.iemss.org/sites/iemss2014/papers/iemss2014_submission_184.pdf.

McStraw, T.C. and others, 2021. An Open-Source Web Application for Regional Analysis of GRACE Groundwater Data and Engaging Stakeholders in Groundwater Management. JAWRA Journal of the American Water Resources Association.

Rodell, M., J.S. Famiglietti, 2002. The potential for satellite-based monitoring of groundwater storage changes using GRACE: the High Plains aquifer, Central US. Journal of Hydrology 263, 245–256. https://doi.org/10.1016/S0022-1694(02)00060-4 available at https://www.sciencedirect.com/science/article/pii/S0022169402000604.

Swain, N.R. and others, 2016. A new open source platform for lowering the barrier for environmental web app development. Environmental Modelling & Software 85, 11–26. https://doi.org/10.1016/j.envsoft.2016.08.003 available at http://www.sciencedirect.com/science/article/pii/S1364815216304625.

Syed, Tajdarul H., James S. Famiglietti, Don P. Chambers, 2009. GRACE-Based Estimates of Terrestrial Freshwater Discharge from Basin to Continental Scales. Journal of Hydrometeorology 10, 22–40. https://doi.org/10.1175/2008JHM993.1 available at https://journals.ametsoc.org/view/journals/hydr/10/1/2008jhm993_1.xml.

Tapley, B.D., Ch. Reigber, 2002. The Grace Mission: Status and Future Plans 4819. Available at https://ui.adsabs.harvard.edu/abs/2002EGSGA..27.4819T.

Voss, K.A. and others, 2013. Groundwater depletion in the Middle East from GRACE with implications for transboundary water management in the Tigris-Euphrates-Western Iran region. Water Resources Research 49, 904–914. https://doi.org/10.1002/wrcr.20078 available at https://onlinelibrary.wiley.com/doi/abs/10.1002/wrcr.20078.

Groundwater is a crucial pillar of water security in the Arab region as it is the primary source of freshwater in more than half of the Arab States, and therefore heavily relied upon. Since the launch of the National Aeronautics and Space Administration Gravity Recovery and Climate Experiment (GRACE) mission in 2002, it has been possible to monitor groundwater storage changes at a large scale using its monthly estimates of total water storage anomalies in equivalent water height. The original mission, which ended in 2017, was succeeded by the GRACE Follow-on mission (GRACE-FO) in 2018 which continues to provide large-scale estimates of total water storage anomalies. While various tools have been developed for processing and visualizing GRACE data, the GRACE Groundwater Subsetting Tool (GGST) is specifically crafted to assist regional stakeholders and decision makers in groundwater resource management.

This training manual was developed to support Arab States access GRACE data available on the Arab Groundwater Knowledge Platform to monitor groundwater storage change. The main objective is to provide a step-by-step instructional guide on the use of the GGST web application to process raw GRACE mission data. The tool eliminates anomalies and enhances resolution of the data, aiming to support the identification and characterization of long-term groundwater storage changes in selected regions. Particularly useful in data-poor areas or regions where trends may be obscured by noise from well data, GGST leverages GRACE mission data to compute and display alterations in groundwater storage through a web-based mapping system, integrating data from both the GRACE and GRACE-FO missions. The results are generated as time series and animated maps of groundwater storage changes.

2400264E