GRACE Groundwater Subsetting Tool

Release 2.0

Prepared by Norm Jones, Sarva Pulla

This manual was issued without official editing.

TABLE OF CONTENTS

CHAPTER

Introduction

Groundwater is a crucial pillar of water security in the Arab region as it is heavily relied upon, and it is the primary source of freshwater in more than eleven Arab States. Recognizing the importance of this resource, The United Nations Economic and Social Commission for Western Asia (ESCWA) has secured funding from 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 impacts 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 on the 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 NASA 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 ended in 2017 and was followed 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 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.

ONE

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).

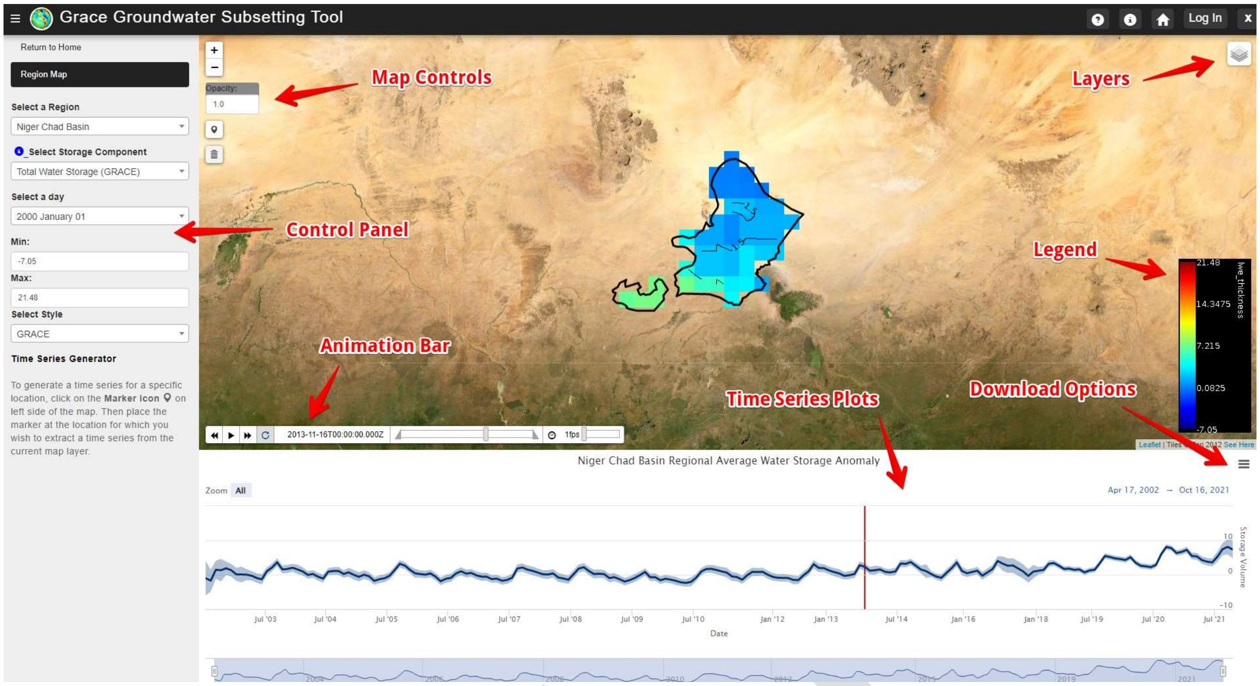
While several tools have been developed for processing and visualizing GRACE data, GGST is designed specifically to support groundwater resource management by regional stakeholders and decision-makers. We accomplish this by carefully processing the raw GRACE data to remove anomalies and improve resolution. This is done 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 we use to process the GRACE and GLDAS data to produce groundwater anomalies on both a global and regional scale is described in detail in the Algorithm section of this document.

Users can access GGST using the Tethys Web Application or by using the API and the associated Google Colaboratory Notebook that makes the API intuitive to use. A brief introduction to these two methods is provided below.

1.1 GGST 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 the GGST as a decision support system to ensure sustainable usage of groundwater. It was developed in the BYU Hydroinformatics laboratory and is now supported by a growing user and developer community. To access the GGST web application, visit https://tethys.byu.edu/apps/ggst/.

Anyone can open the app to view the currently uploaded regions and download the time series plots. View the screenshot below to see how to manipulate the map and download the data. Users can change the storage component displayed and the color 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 NETCDF file raster that is displayed but this can be downloaded using the 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 feel free to reach out to BYU or ESCWA's team if you would like a region uploaded.

The GGST app can be accessed in the following locations:

- **SERVIR West Africa Portal**: Official Tethys portal hosted by the SERVIR Science Coordination Office (SCO) for the West Africa Hub (https://tethyswa.servirglobal.net/apps/).
- **BYU Portal**: A Tethys portal hosted by Brigham Young University for experimental web applications developed by the BYU Hydroinformatics Lab (https://tethys.byu.edu).
- **GEOGloWS Portal**: A portal associated with the UN GEOGloWS global streamflow forecasting initiative (https://apps.geoglows.org/apps/).
- **ESCWA Cloud**: The GGST app is also available now on ESCWA's cloud and it is used to process GRACE groundwater data for the Arab Region and 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/

1.2 API 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 application programmer interface (API). The advantages of the API are that you can retrieve data about 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 the API on your own, but we recommend using the Google Colaboratory Notebook hosted on GitHub which is de- signed 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 that is a component of the Google Drive environment.

TWO

COMPUTATIONAL ALGORITHM

The GRACE Groundwater Subsetting Tool (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 3-degree resolution. This map is then down-scaled using a mass conservation algorithm to 0.5-degree resolution and made available for download in the netCDF multidimensional raster format.

2.1 Derivation of Groundwater Dataset

The groundwater component of the GRACE raw data can be separated using a mass balance approach, with NASA's Global Land Data Assimilation System (GLDAS) models to compute the surface water component of the data. To compute total surface water storage, we sum the components of the GLDAS models that represent surface water storage and then subtract this total from the GRACE dataset to estimate a groundwater storage anomaly dataset.

This GSST application uses four sets of data:

- The GRACE TWSa dataset
- The GLDAS canopy storage dataset (CAN)
- The GLDAS snow water equivalent (SWE)
- The GLDAS soil moisture (SM)

To compute the groundwater storage anomaly (GWa), we use three components of the GLDAS models: CAN, SWE, and SM. We convert each GLDAS component to an anomaly format by subtracting the mean centered on values from 2004 to 2009 and then average across the three GLDAS models to produce a component anomaly dataset: CANa, SWEa, and SMa. We use the standard deviation from the three GLDAS models to help estimate uncertainty.

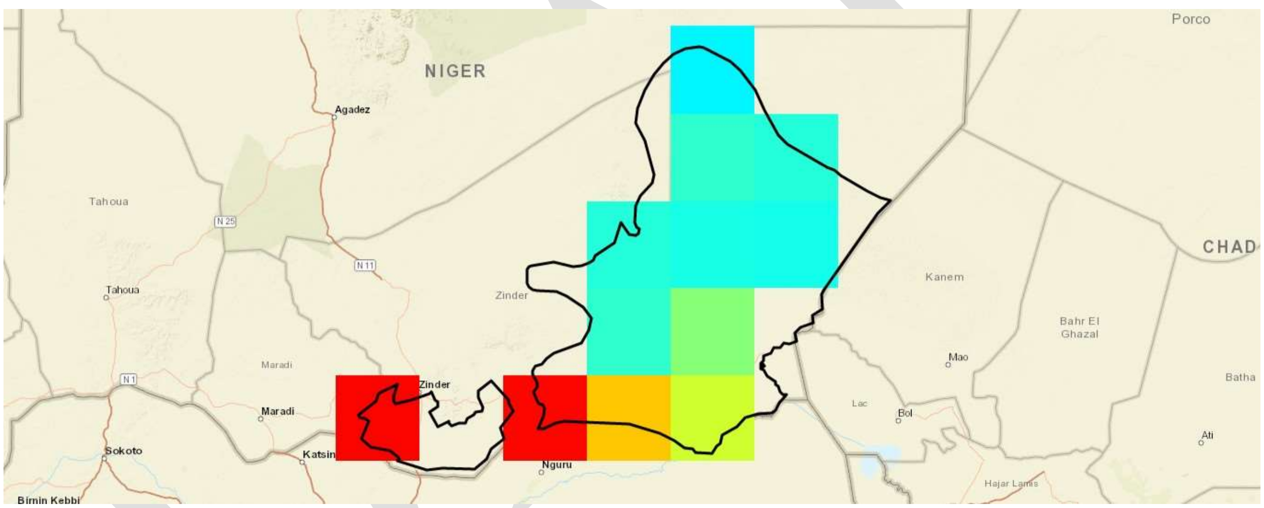
We download GLDAS files, format them as netCDF, and store them locally. Normally, the data is acquired in a gridded format with a 1-degree latitude by 1-degree longitude resolution, which we then convert to a 0.5-degree resolution. This conversion is performed by an area-weighted average of the four 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 the TWSa and the sum of the surface water component anomalies.

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

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

2.2 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 cell centers within the defined boundary and calculate the average storage anomaly for each of the components: TWSa, SWEa, CANa, and SMa resulting in a time series from 2002 to the present for each component on a monthly time step. The figure below 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.



2.3 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, we combine the uncertainty estimates from both the GRACE and GLDAS 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.

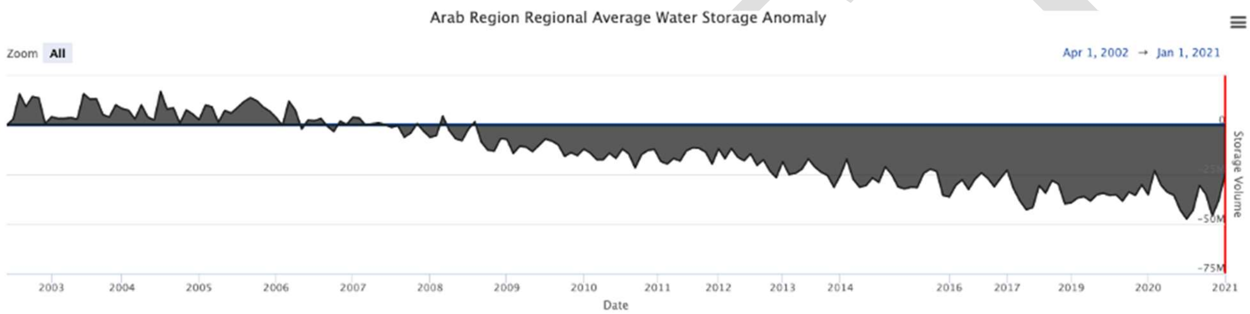
2.4 Storage Depletion Curve

The 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, we sum the GWSa over time to determine changes in groundwater storage volume over time for the region. These data show if a region is depleting storage in the region, or if groundwater is recharging in the region thereby providing valuable information relative to groundwater sustainability.

Here is an illustration of Northern Africa and the Arabian Peninsula from 2002 - 2021. It shows that the groundwater in that region has been depleting since early 2009 and onward.



2.5 Limitations

GRACE data come with limitations that users need to know and understand. The data are 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 provide 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 the GGST is to draw general trends in aquifers rather than selecting a placement of a well.

We recommend that, whenever possible, these 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.

2.6 Software Availability

The GGST web application was created using Tethys Platform, developed in the BYU Hydroinformatics Laboratory. See Section 1.1 for more information on this web application and links to use it.

CHAPTER

THREE

ADDING AND DELETING REGIONS

This section describes how to upload new regions to the GGST app. When a new region is uploaded, it is automically processed and the storage components including subsetting netCDF files and storage time series are computed for the region and stored with the region for visualization in the app. 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.

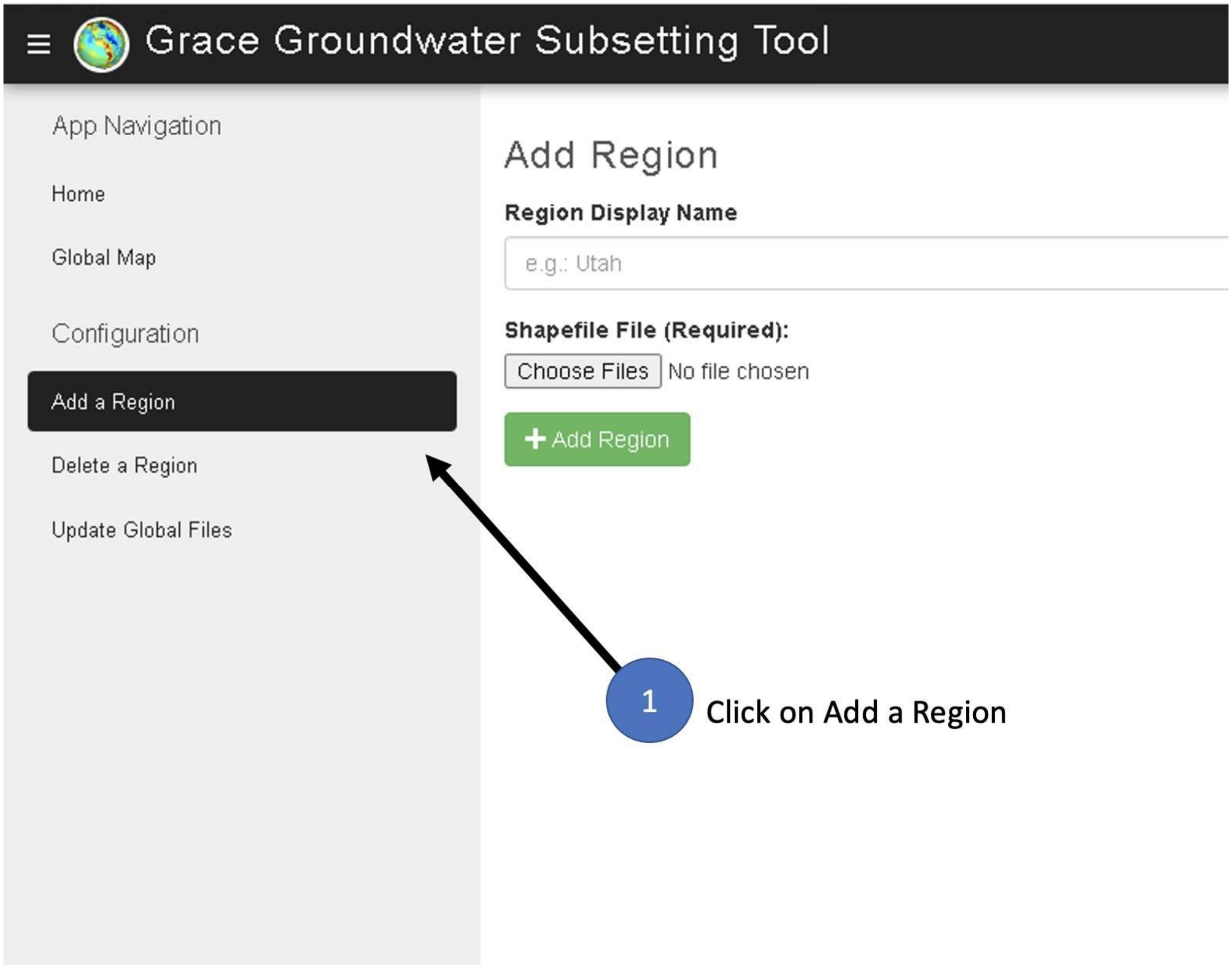
3.1 Uploading a Region

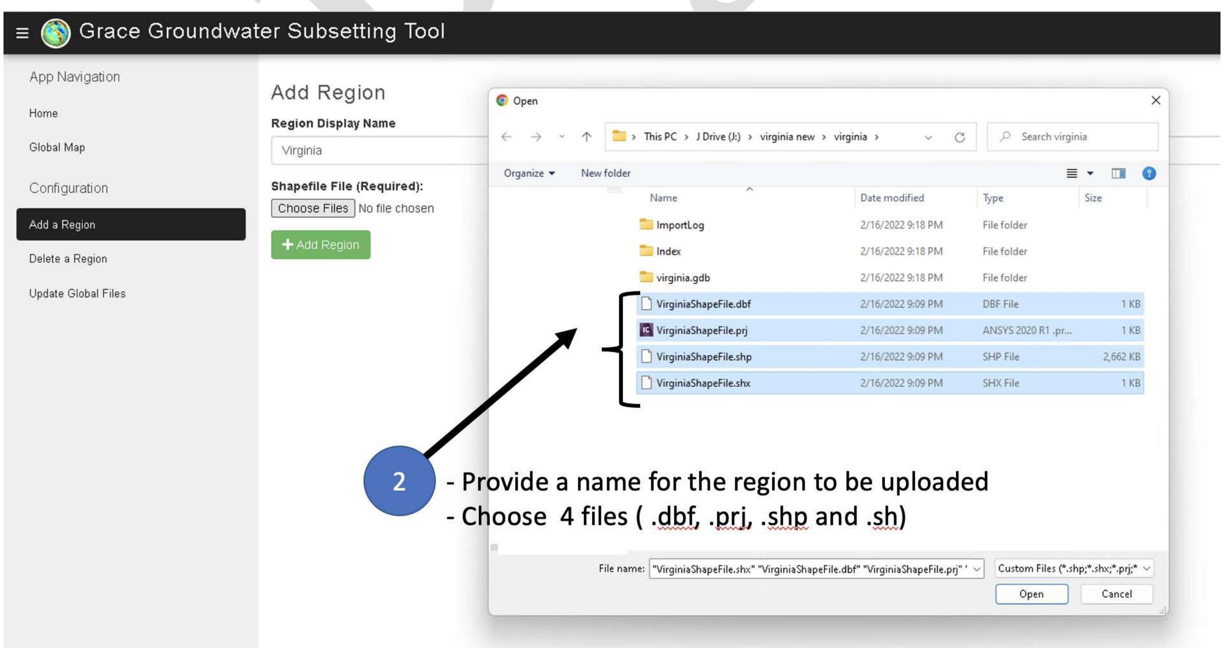
As described in section 1.1 previously, the GRACE Groundwater Subsetting Tool web application is hosted on four different Tethys portals.

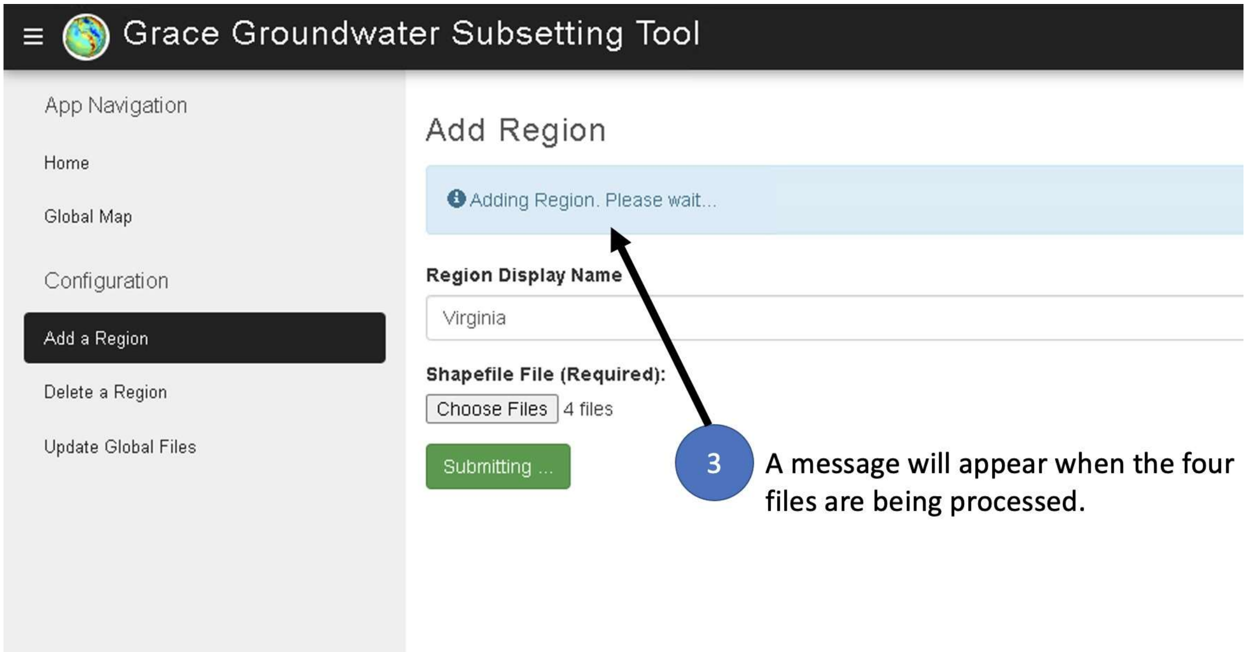
To upload regions on the application, visit the portal of your choice and log in. Without logging in you can see the App Navigation pages: Home and Global Map. These allow you to view previously uploaded regions and create time series graphics for any singular point on the globe. Once you log in with 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.

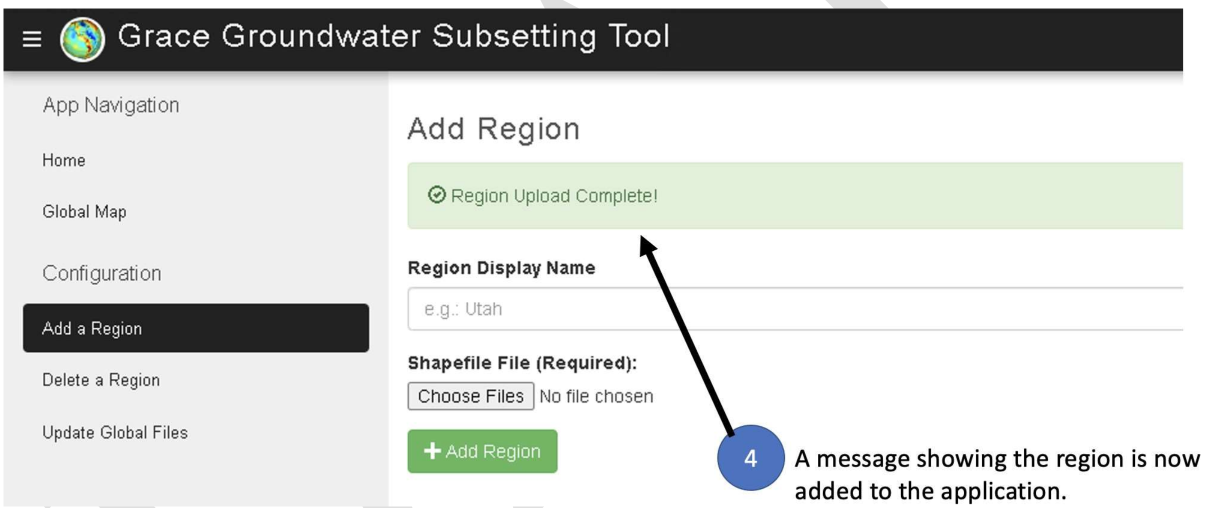
To add a new region, first prepare a shapefile for the region consisting of four files: *.shp, *.dbf, *.prj, and *.shx. The projection for the shapefile should be **EPSG:4326 - WGS 84**. The four files should not be zipped together.

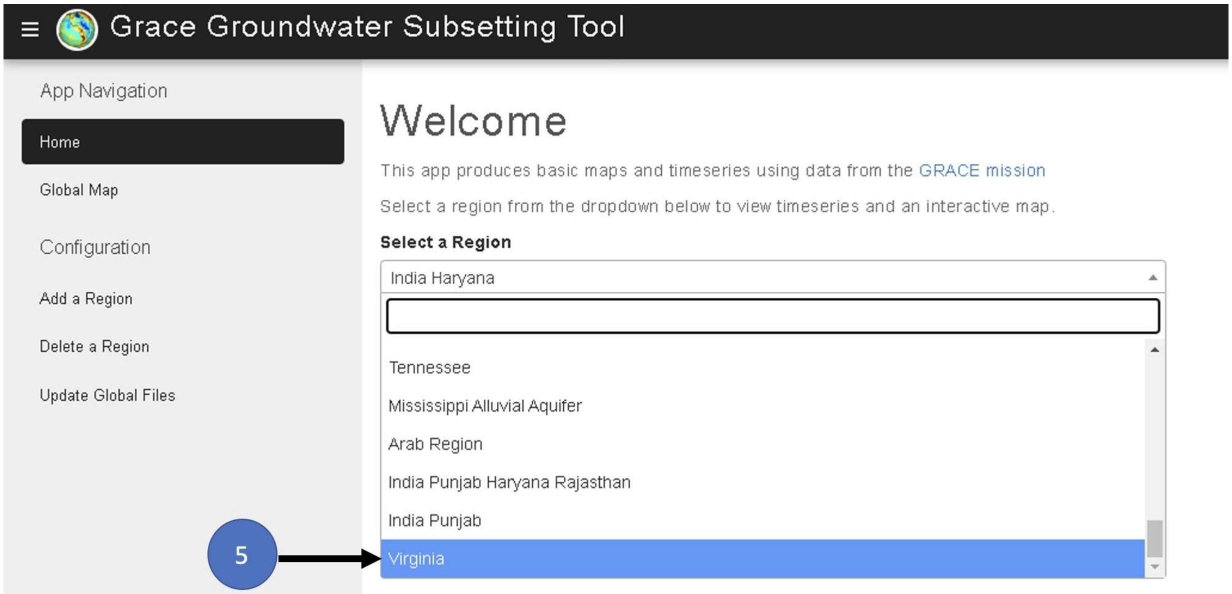
Please refer to the following images as a visual guide:


Grace Groundwater Subsetting Tool
App Navigation
Home
Global Map
Configuration
Add a Region
Delete a Region
Update Global Files
Add Region
Region Display Name
e.g.: Utah
Shapefile File (Required):
Choose Files
No file chosen
Add Region
1
Click on Add a Region


Grace Groundwater Subsetting Tool
App Navigation
Home
Global Map
Configuration
Add a Region
Delete a Region
Update Global Files
Add Region
Region Display Name
Virginia
Shapefile File (Required):
Choose Files
No file chosen
Add Region
Open
This PC › J Drive (J:) › virginia new › virginia
Search virginia
Organize
New folder
Name
Date modified
Type
Size
ImportLog
Index
virginia.gdb
VirginiaShapeFile.dbf
VirginiaShapeFile.prj
VirginiaShapeFile.shp
VirginiaShapeFile.shx
File name:
Open
Cancel
2
- Provide a name for the region to be uploaded
- Choose 4 files (.dbf, .prj, .shp and .sh)


Grace Groundwater Subsetting Tool
App Navigation
Home
Global Map
Configuration
Add a Region
Delete a Region
Update Global Files
Add Region
Adding Region. Please wait...
Region Display Name
Virginia
Shapefile File (Required):
Choose Files 4 files
Submitting ...
3
A message will appear when the four files are being processed.

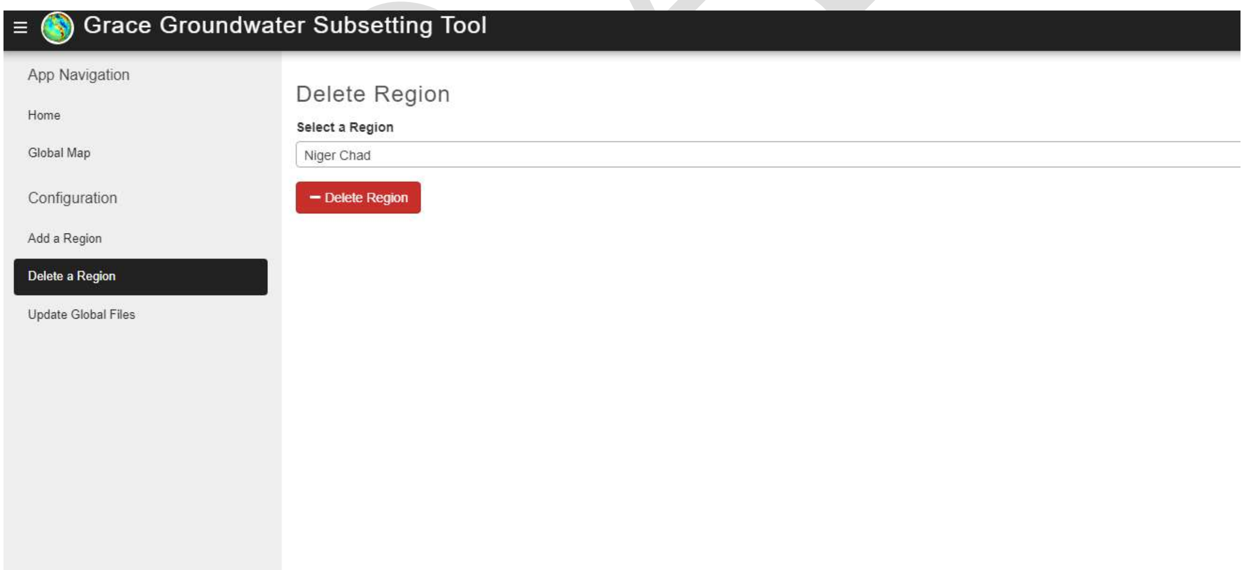

Grace Groundwater Subsetting Tool
App Navigation
Home
Global Map
Configuration
Add a Region
Delete a Region
Update Global Files
Add Region
Region Upload Complete!
Region Display Name
e.g.: Utah
Shapefile File (Required):
Choose Files No file chosen
Add Region
4
A message showing the region is now added to the application.



The region can be accessed through the Home ribbon

3.2 Deleting a Region

Deleting a region is very simple. Proceed to the Delete a Region page. Select the region from the drop-down menu and hit the delete button. A message will display when the deletion has been completed.



CHAPTER

FOUR

APPLICATION PROGRAMMING INTERFACE (API)

The Python API for the GGST allows users to retrieve groundwater information about a point or region without having administrative privileges to the GGST web application. The 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 the third section of this documentation on how to obtain the said token. The API can be implemented in many ways using a variety of coding languages and platforms. We have provided an example implementation using the Python code language in a Google Colaboratory notebook. If you choose to use Python to call the API, we recommend the xarray and geopandas Python packages be used to process your data. The former helps in visualizing and interacting with the raw netCDF data returned while the latter helps in uploading the shapefile(s) for the subsetting.

Before we use the API in the Google Colaboratory notebook, let us explore each of the four GGST API methods in the following section.

4.1 API 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://tethys-staging.byu.edu/apps/[parent-app]/api/[MethodName]/?param1=value1&param2=value2&. . . paramN=valueN

To test the API, the user will need a zip file of the region of interest. We have provided a set of sample files in the appropriate format. You may use your own files if you choose so.

`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:

1. **The getStorageOptions**

Parent application	GGST
Supported Methods	GET
Returns	A JSON object with a list of storage options
Parameters	There are no parameters for the getStorageOptions function

Follow this link to inspect the JSON returned which lists the list of the storage options available. https://tethys-staging.byu.edu/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**

Parent application	GGST			
Supported Methods	GET			
Returns	A JSON object with a timeseries for a given point			
Parameters	**Name**	**Description**	**Valid Value**	**Re-quired**
	Longi-tude	long in WGS 84 Proj	Any value on land with the GRACE Explorer Doman (-60,180)	Yes
	Latitude	lat in WGS 84 Proj	Any value on land with the GRACE Explorer Doman (-60,90)	Yes
	stor-age_type	Storage type of interest	One of the abbreviated values from the first function. eg. grace, sw, sm or gw	Yes

Open the following example link to call the API and inspect the JSON object returned (results will appear in a new window). This is a return for obtaining points values https://tethys-staging.byu.edu/apps/ggst/api/getPointValues/?latitude=20.7&longitude=80.2&storage_type=gw for an example.

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 Colab Notebook for further instructions. See details below on how to obtain one.

3. The getRegionTimeseries

Parent application	GGST			
Supported Methods	POST			
Returns	A JSON object with area of the region, depletion time series, error range timeseries 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. eg. grace, sw, sm or gw	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

Example query: files = {'shapefile': ("response.zip", uploaded["".join(uploaded)],'application/zip')} re-on_timeseries_request = requests.post("https://tethys-staging.byu.edu/apps/ggst/api/getRegionTimeseries/", eaders={"Authorization": f"Token {api_token}"}, data = {"name":"api_test", "storage_type": "tws"}, files=files) esponse (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

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

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://tethys-staging.byu.edu/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()

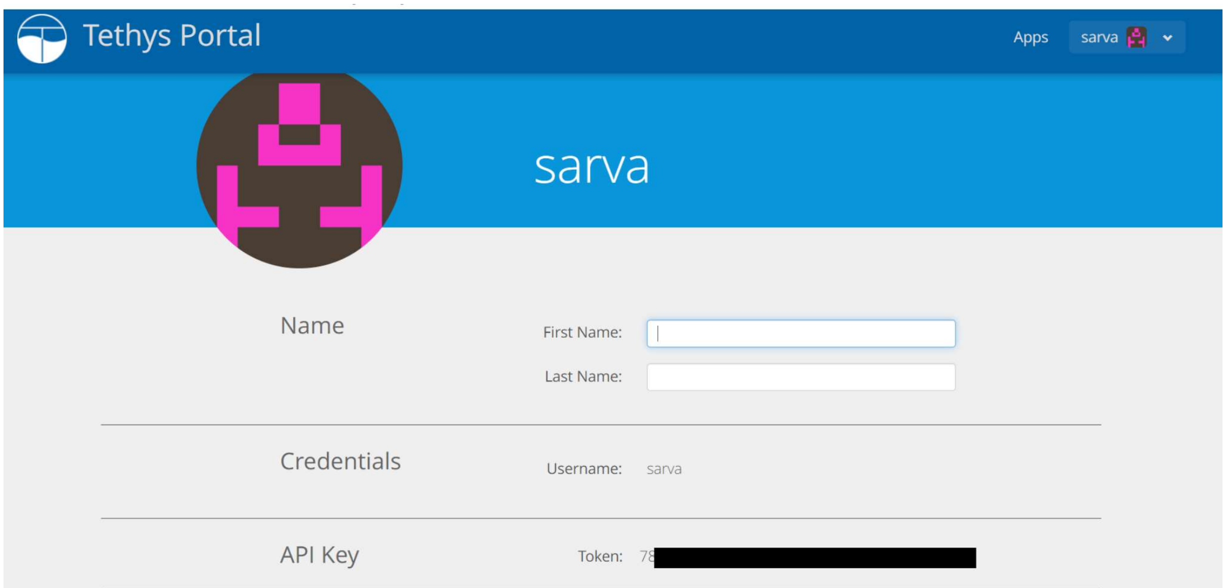
Result will be a folder with nc files.

4.2 Obtaining an Authentication Token

The last two functions of the API require an authentication token. To obtain one, you will need to sign up for an account on BYU Tethys Portal (https://tethys-staging.byu.edu/apps/). 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 the User Settings to reveal the API key.

The authentication token or API key will be in the third section.



For privacy reasons, we have hidden the remaining characters of this user's token.

It is also possible to request an authentication token directly from an administrator. We recommend using the sign-up method as it is faster.

4.3 GGST API Google Colaboratory Notebook

We have provided an example of calling the GGST API using the Python coding language in a Google Colaboratory Notebook which you can open 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 we described above. 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.

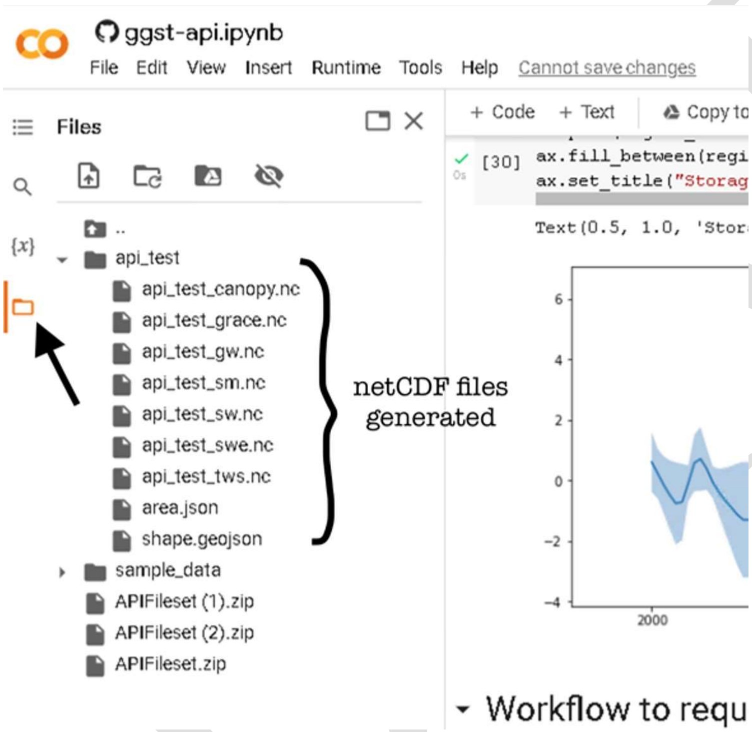
The code is divided into six sections designed to help the user understand how to call each of the four functions and how to plot and visualize them.

- **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, three portals are available: the Tethys staging, Tethys main and the Tethys West Africa. The first two are maintained by the Brigham Young University Hydroinformatics Laboratory and the last by the SERVIR program in West Africa. The Tethys staging portal is the testing ground for web applications developed by the BYU Hydroinformatics Laboratory before committing them to the two main portals.
- **Function 1: getStorageOptions**
  This cell lists all the available options and how to properly declare them in the appropriate cell.
- **Function 2: getPointValues**
  The user types in latitude and longitude coordinates and selects the desired storage option from a drop-down menu. The next several cells will create a dataframe, chart the timeseries, and plot a graph with estimated error bars.

- **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 you through inputting that additional information. First, you will be asked for your API token which must match your declared portal to work. Second, you will be asked to give your region a name that will be used in naming the files. Lastly, you will be asked to upload a zipped shapefile of the region of interest. This should contain four files (a .shp, .shx, .prj, and .dbf) zipped in a single folder.
- **Function 3: getRegionTimeseries**
  Asks for your desired storage option using a drop-down menu, calls the API, then displays an interactive table and graph of the data returned.
- **Function 4: getRegionZipfile**
  Calls the API and returns a set of netCDF files which can be accessed from a tool bar on the left side of the screen as pictured below.

Visual guide on netCDF files:



This table elaborates on each of these files and their naming conventions:

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

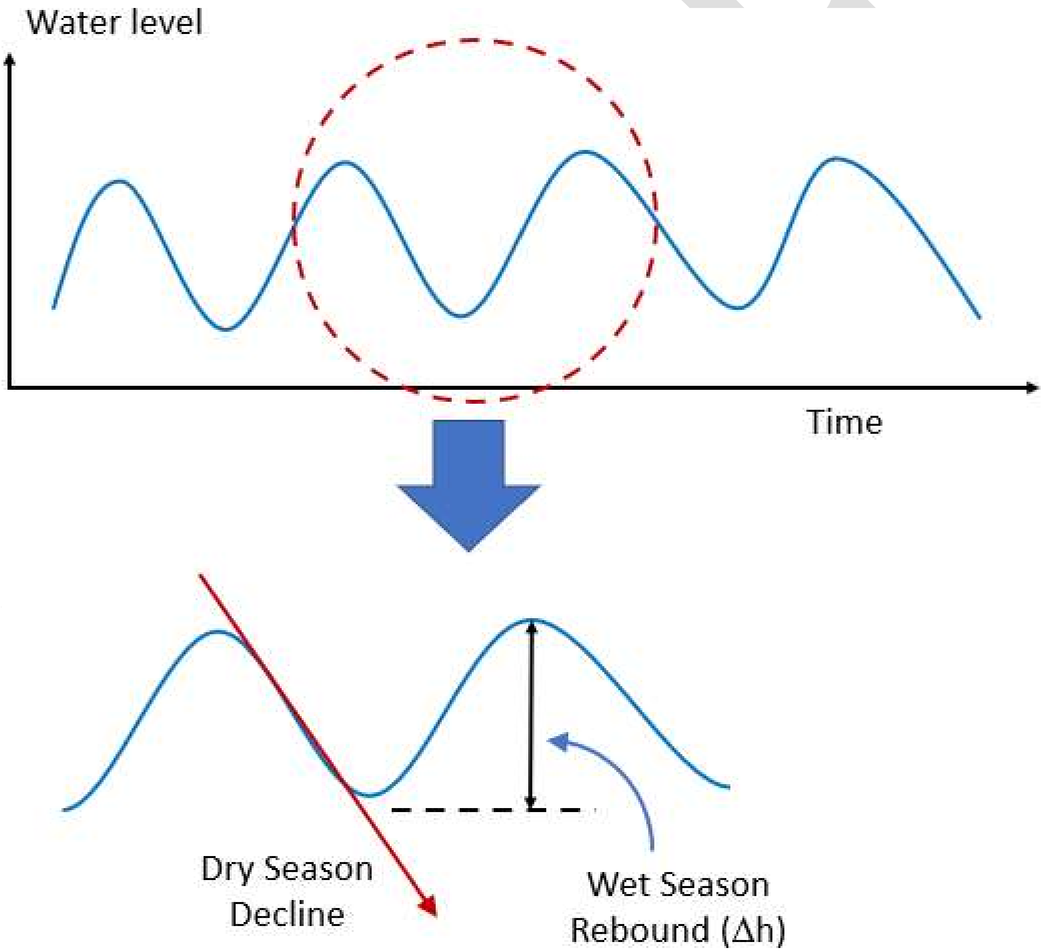
*To learn more about how this is calculated please visit our Computational Algorithm page. This section will also help you create a dataframe, plot your data, and visualize your data on an animated map.

CHAPTER

FIVE

THE WATER TABLE FLUCTUATION METHOD

In addition to obtaining the GRACE-derived groundwater storage anomaly, it is possible to analyze 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 below, 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, we can estimate the recharge as follows:

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

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.

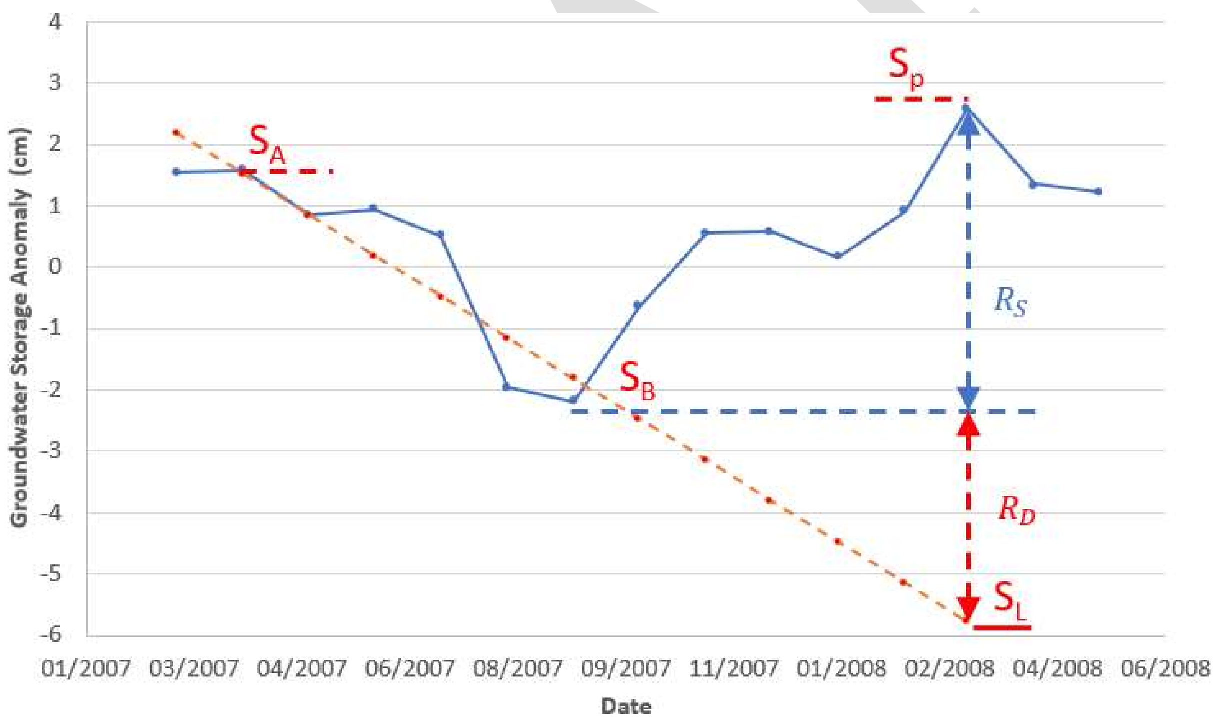
The storage coefficient is necessary because the water level rise in the surrounding aquifer occurs in the fractional void space and the storage coefficient converts it to the appropriate liquid water equivalent component in the [length]/[time] infiltration rate units used by recharge. If we perform this analysis using the groundwater storage anomaly curve derived from GRACE, we do not need to use a storage coefficient as the anomaly is already in liquid water equivalent form and we can directly estimate the recharge as:

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

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

5.1 Methods for Estimating Recharge Component

There are two general approaches for determining the height of the rise associated with recharge:



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

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

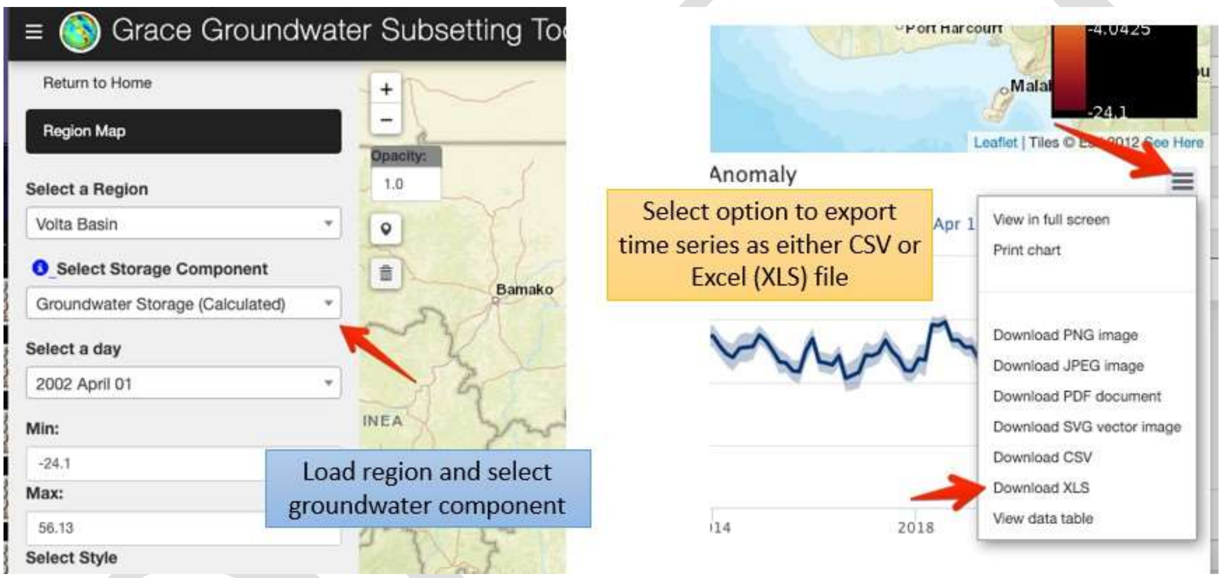
Another method is to assume that the groundwater decline because of pumping and discharge continues at the same rate in the wet season and therefore the rise should be computed from a linear extrapolation of the declining line as follows:

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

The recharge rates extracted from these two equations could be considered a low and a high estimate, although in our experience method 1 seems to be the most accurate. An example of applying the WTF method to estimate recharge in Southern Niger can be found: Evaluating Groundwater Storage Change and Recharge Using GRACE Data: A Case Study of Aquifers in Niger, West Africa (Barbosa et al., 2022).

5.2 Downloading the Water Level Time Series from the GGST App

To apply the WTF method to estimate recharge on GRACE data, one must first download the groundwater storage anomaly time series from the GGST app. To do so, first load the region and select the Groundwater Storage (Calculated) storage component and then 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.



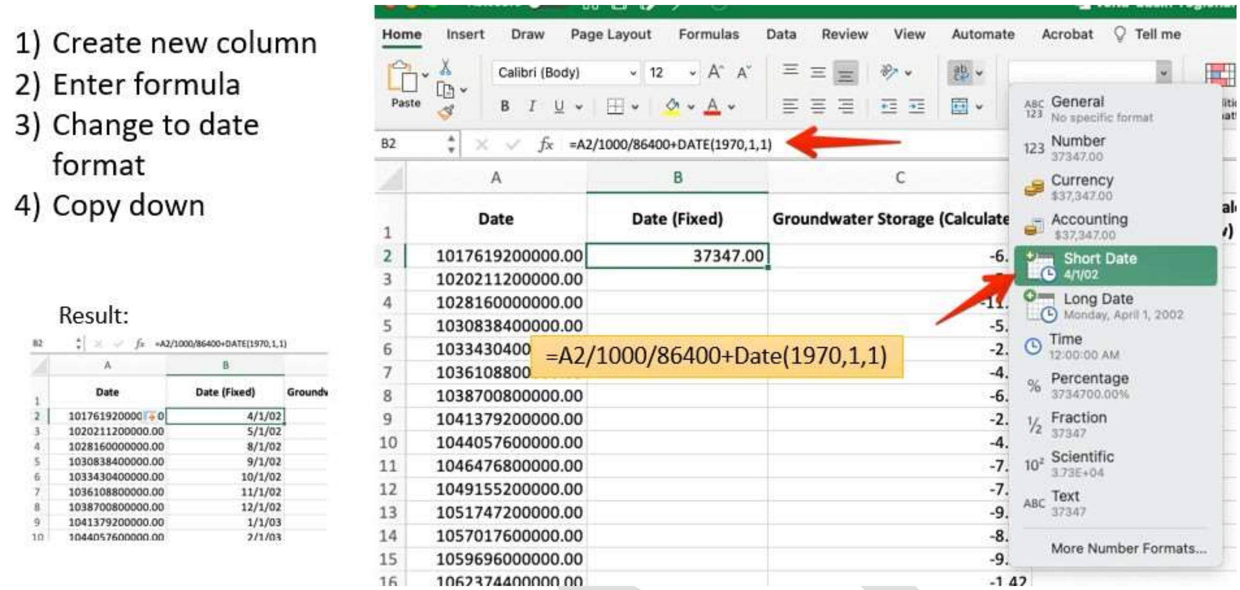
The storage anomaly chart is created, displayed, and downloaded using the HighCharts plugin. The format of the resulting downloaded file is as follows:

	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

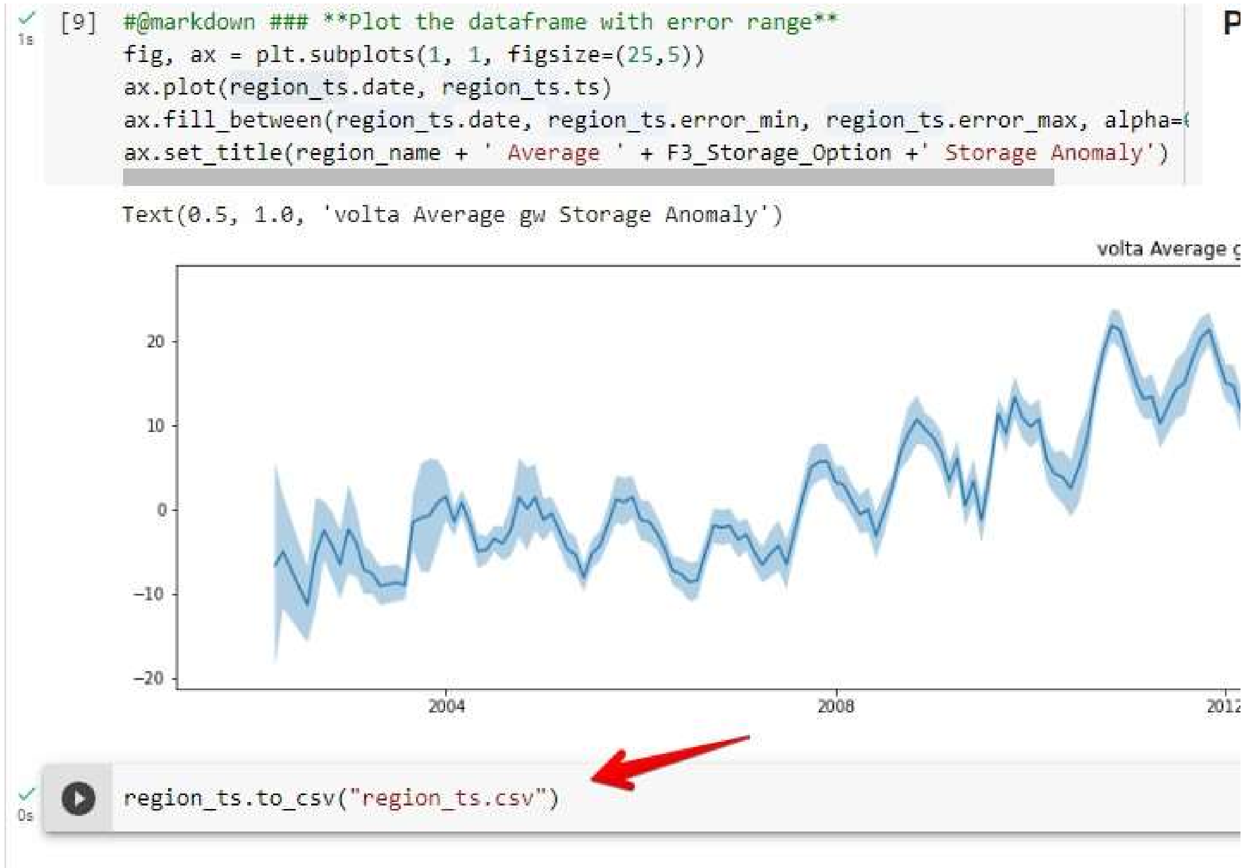
milliseconds since january 1, 1970

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

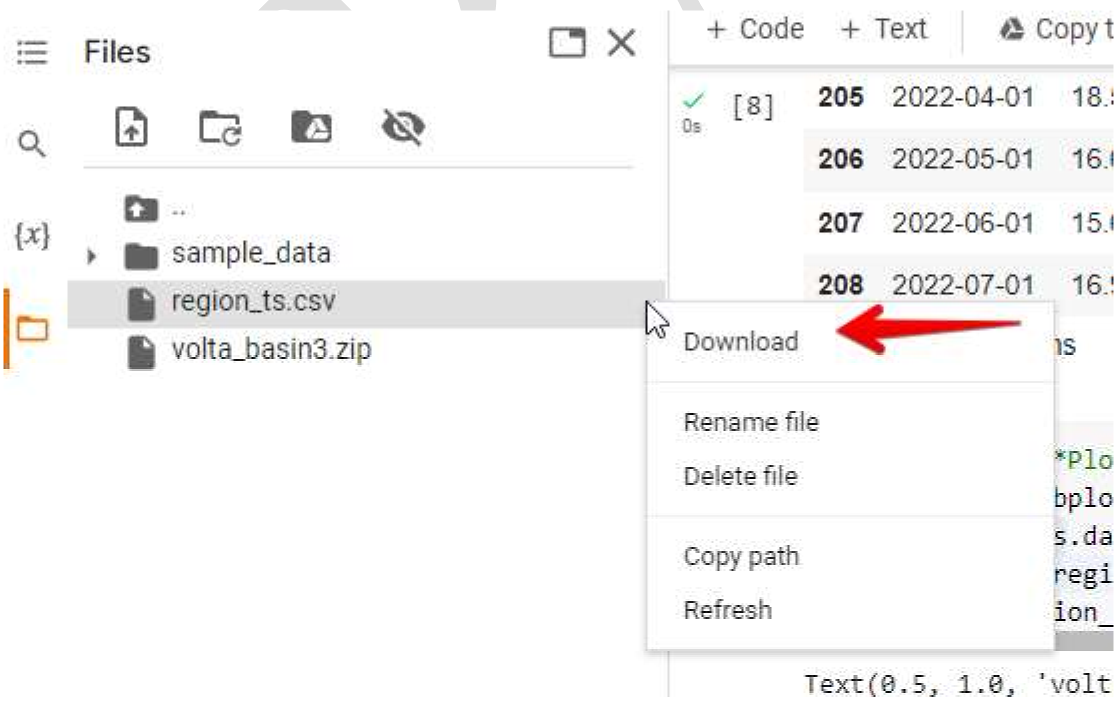


5.3 Downloading the Water Level Time Series from the API Google Colaboratory Notebook

You can also download the time series directly from the sample 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 data frame containing the time series to a CSV file.



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



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

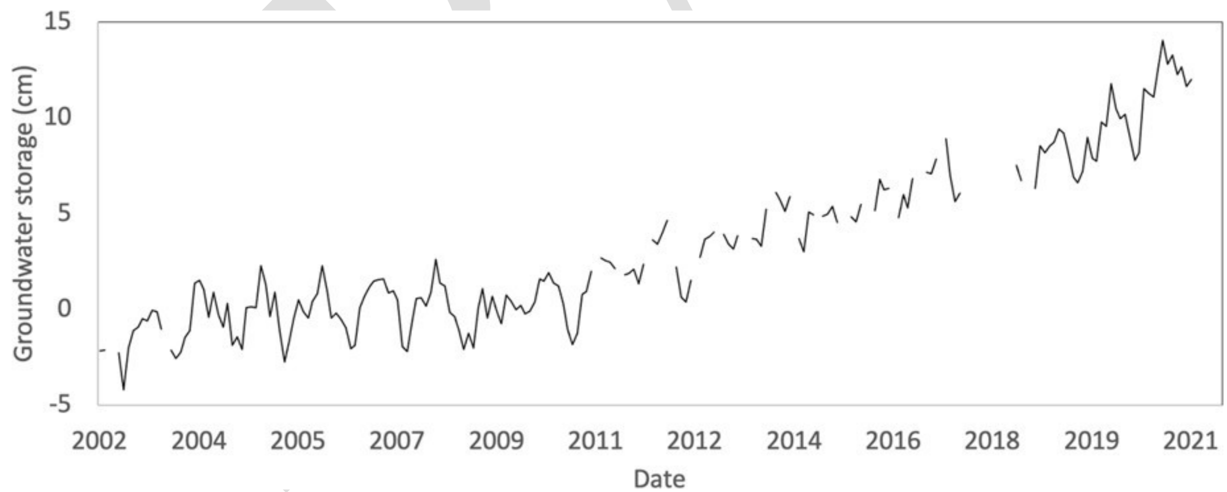
	A	B	C	D	E	F
1		date	ts	error_min	error_max	
2	0	4/1/2002	-6.66	-18.978	5.658	
3	1	5/1/2002	-5.033	-11.749	1.683	
4	2	8/1/2002	-11.323	-15.81	-6.835	
5	3	9/1/2002	-5.273	-11.971	1.426	
6	4	10/1/2002	-2.464	-5.961	1.034	
7	5	11/1/2002	-4.24	-8.214	-0.267	
8	6	12/1/2002	-6.551	-10.684	-2.419	
9	7	1/1/2003	-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	

5.4 Gaps in the GRACE Data

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

	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	

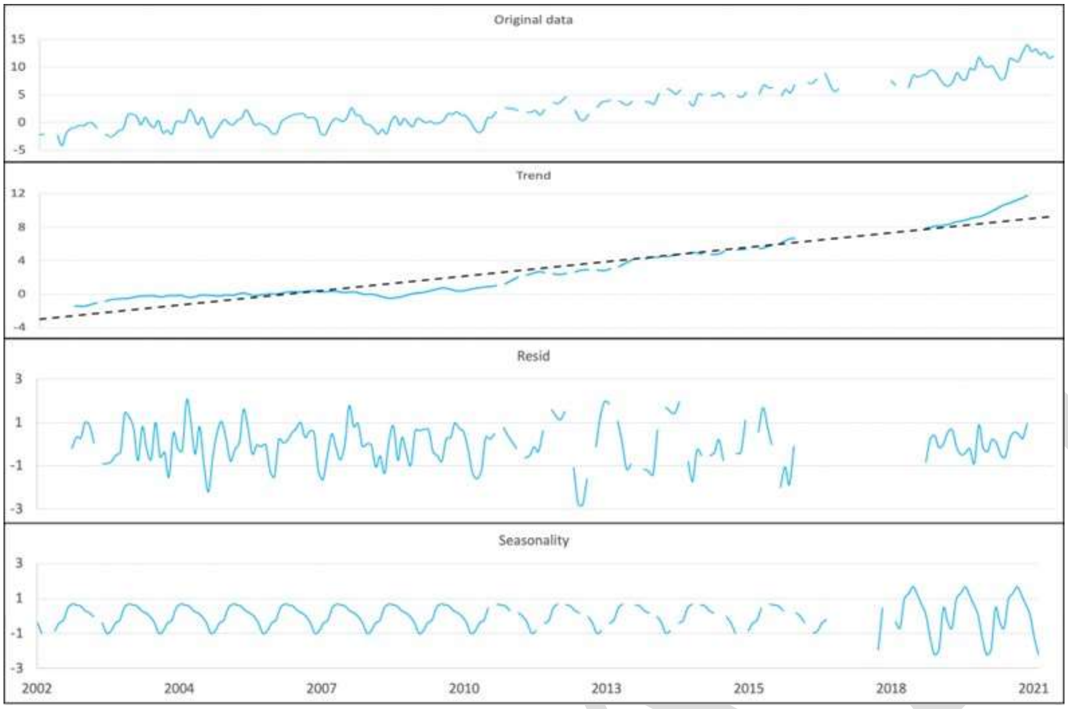
This is because there were periods when the GRACE satellites did not produce usable data. The largest gap is a 12-month period in 2017-2018 between the end of the original GRACE mission in 2017 and when the subsequent GRACE-FO satellites were launched and became operational in 2018. Here is a sample 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 our case months, with the residual component being the difference between the monthly average (seasonal component) and the actual monthly measurements. With this approach, we decompose the GWSa time series into three components: the trend, the seasonal, and the random components:

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

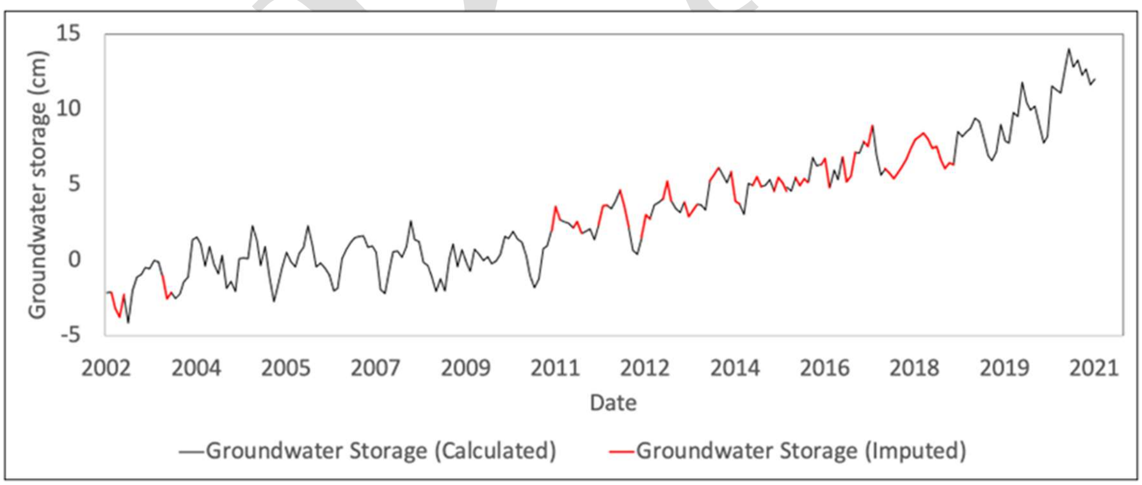
Where Y[t] is the GWSa, T[t] is the GWSa trend, S[t] is the seasonal GWSa component, and e[t] is the residual GWSa component. The decomposition components for the data shown above are as illustrated here:



To impute the missing data, we 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:

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

The following figure shows the original time series in black, with imputed values in red:



5.5 Data Imputation Tools

To assist users in applying the statsmodel method described above to impute gaps in the GRACE data, we have implemented the Python code to perform the imputation in a Google Colab notebook whose link is below. After launching the notebook, follow the instructions in the code.

Before running the code, you 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 you can copy and paste from the full CSV and then save as a separate CSV file ("base_file.csv" for example).

	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	

Furthermore, the code will automatically detect small gaps, but the large gap from 2017-2018 must be identified by creating empty rows in the file as follows:

	A	B	C	D
160	4/1/2017	7.554		
161	5/1/2017	3.686		
162	6/1/2017	3.483		
163	7/1/2017			
164	8/1/2017			
165	9/1/2017			
166	10/1/2017			
167	11/1/2017			
168	12/1/2017			
169	1/1/2018			
170	2/1/2018			
171	3/1/2018			
172	4/1/2018			
173	5/1/2018			
174	6/1/2018	7.4		
175	7/1/2018	6.899		
176	10/1/2018	18.179		
177	11/1/2018	17.9		
178	12/1/2018	19.076		

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

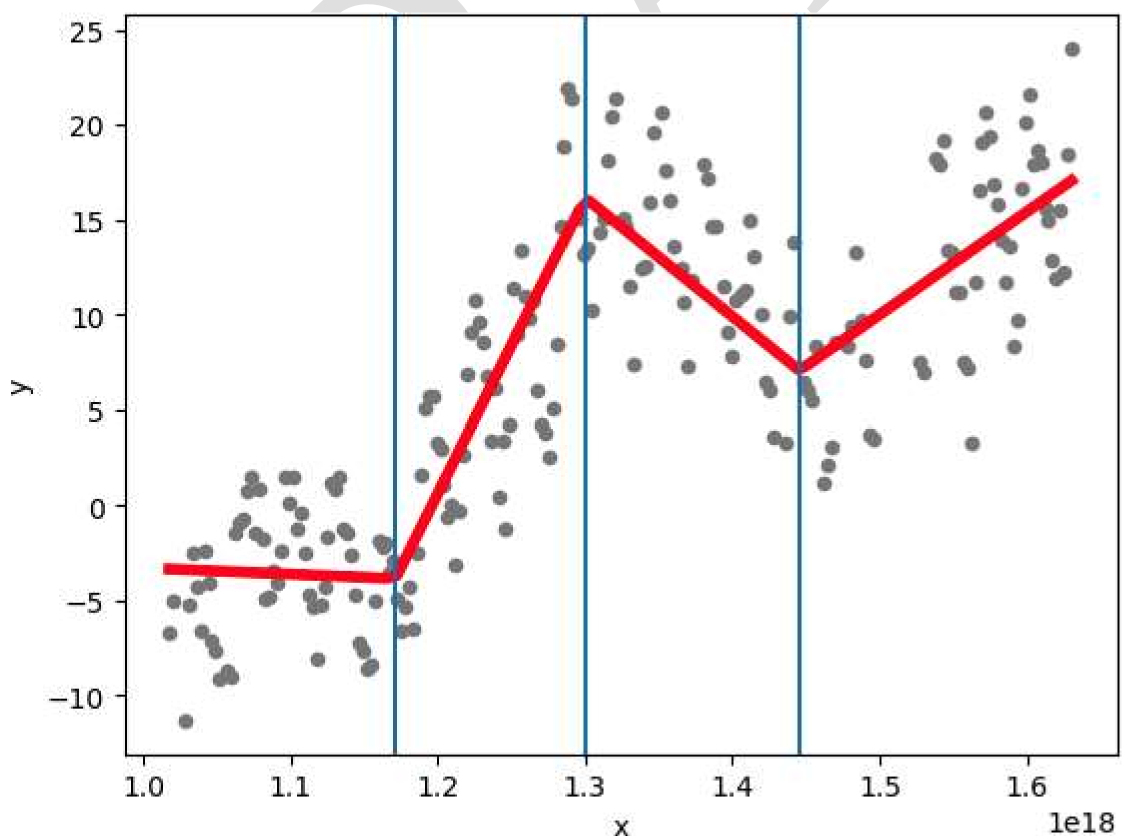
Here is a sample file you can use with the script: [west-gwsa-raw-clean.csv](https://ggst.readthedocs.io/en/latest/_downloads/2a3c7527df3de835c68ed6347fa4496f/west-gwsa-raw-clean.csv)

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

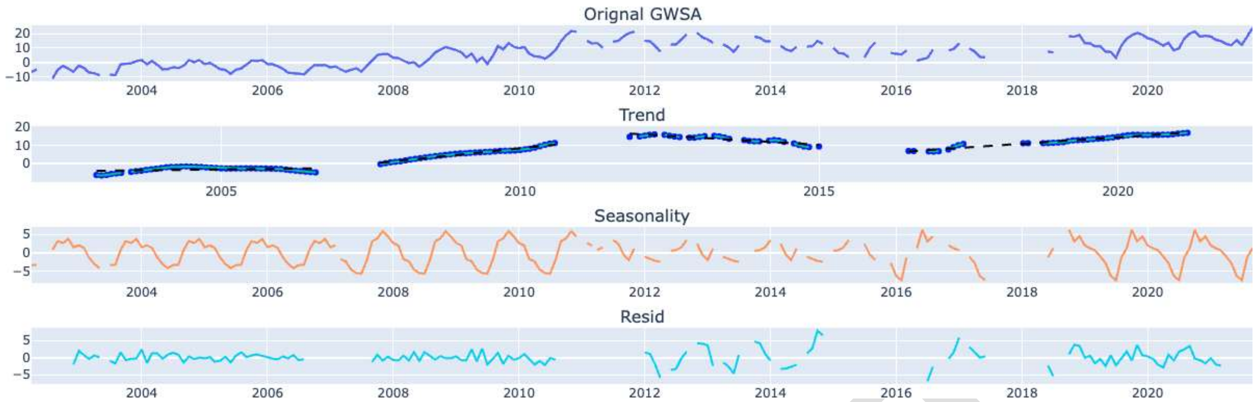
5.6 Multi-Linear Trend Analysis

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

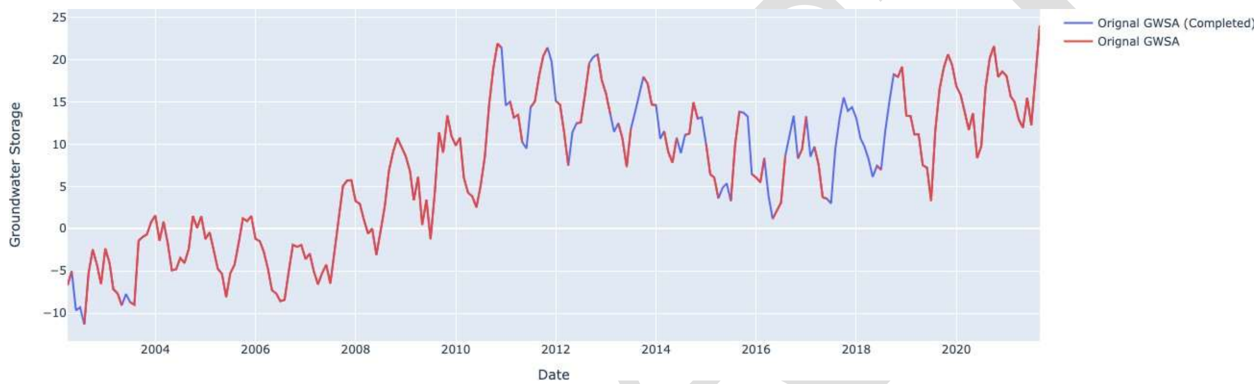
However, many data sets exhibit multiple linear trends. For this dataset, there are four distinct trends. The Python script has an option to perform a multi-linear regression analysis. For this dataset, we set the number_breakpoints variable to 3, and run a multi-linear regression algorithm that fits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



Note that 3 interior breakpoints result in four linear trends. This option results in the following trends:



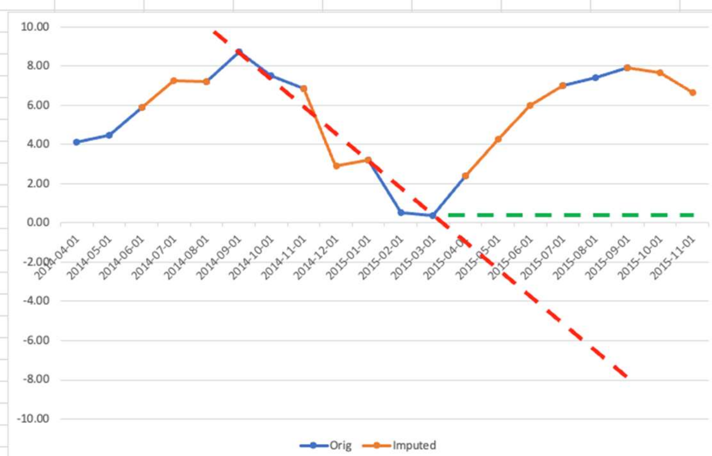
And finally, the gap imputation with 4 trend lines results in the following:



5.7 Data Processing Examples

Once the gaps have been filled, the last step is to plot and analyze 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.

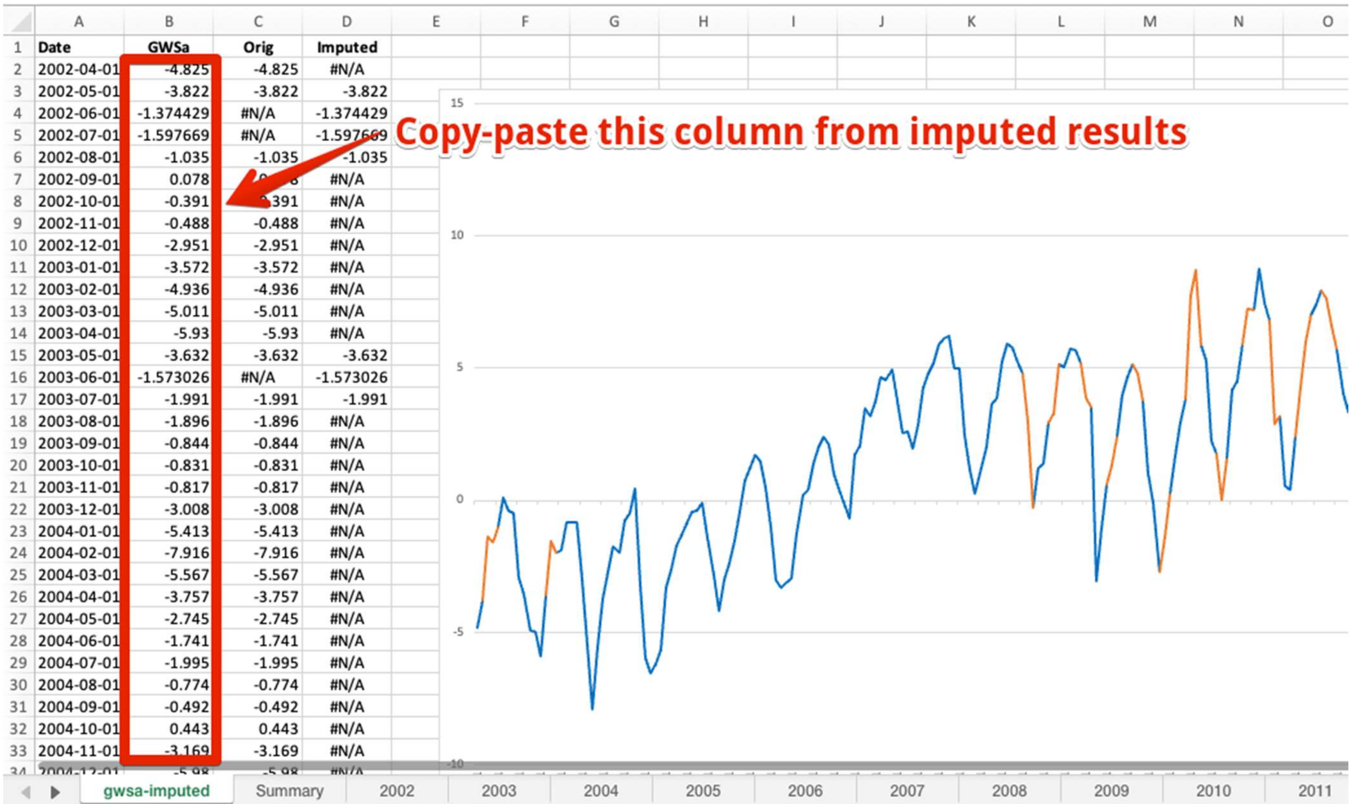
	A	B	C	D	E	F	G
1	**Year:**	**2014**					
2							
3	**Date**	**GWSa**	**Orig**	**Imputed**			
4	2014-04-01	4.14	4.14	#N/A			
5	2014-05-01	4.49	4.49	#N/A		SP	8
6	2014-06-01	5.89	5.89	5.89		SB	0.2
7	2014-07-01	7.24	#N/A	7.24		SL	-8
8	2014-08-01	7.20	7.20	7.20			
9	2014-09-01	8.74	8.74	#N/A		RS	7.8
10	2014-10-01	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							



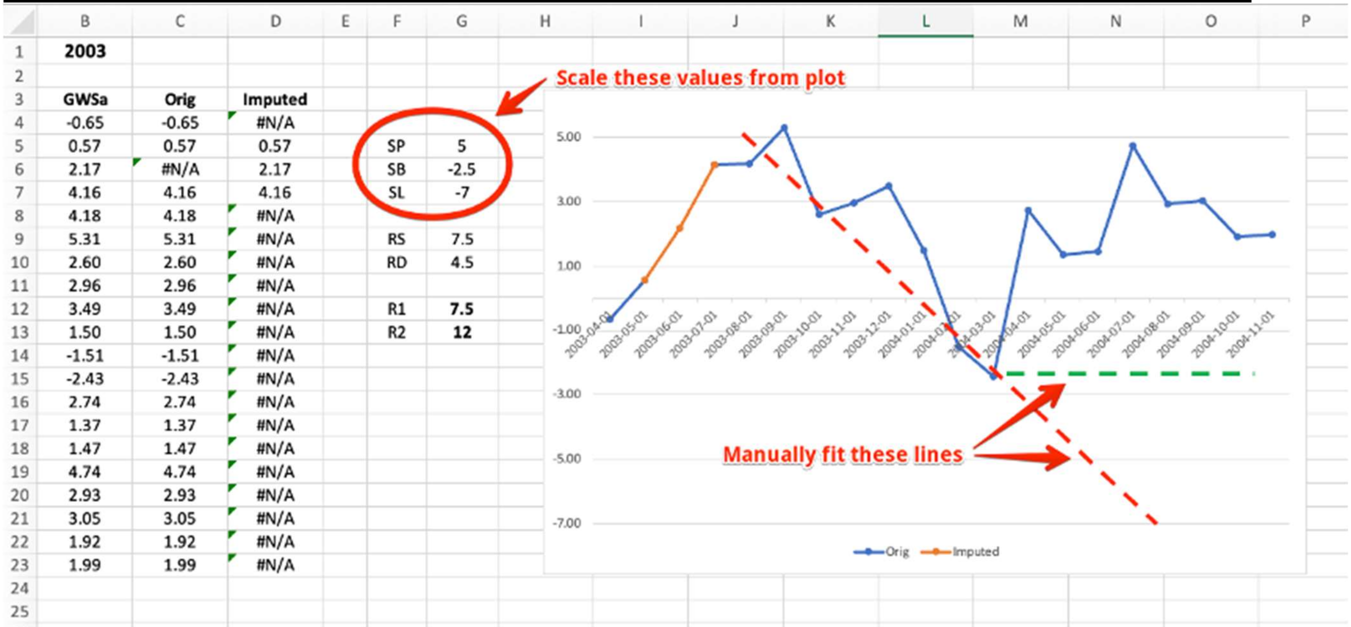
The following Excel file illustrates how to examine and process each season of data from a GRACE-derived and imputed groundwater storage anomaly time series: west-gwsa-wtf.xlsx.

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 here. Note that the imputed values have more digits than the original values. The formulas in columns C & D separate the imputed data in column B to allow a multi-colored plot where the original imputed sections can be clearly visualized.

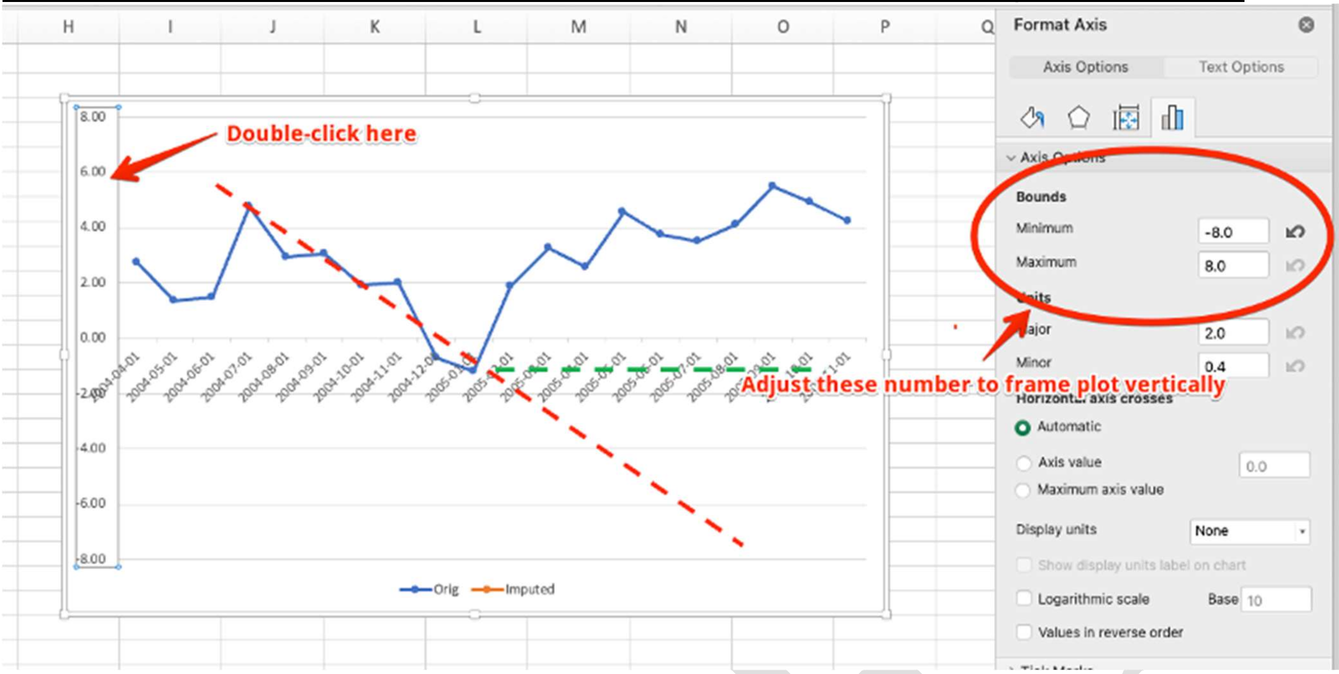


At this point you can browse through each of the tabs for the years starting in 2002. On each page, the seasonal values are automatically pulled from the main sheet using a VLOOKUP formula. For each page, manually adjust the red and green lines to fit the descending branch and the base. Then manually 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.



	GWSa	Orig	Imputed		
1	2003				
2					
3	GWSa	Orig	Imputed		
4	-0.65	-0.65	#N/A		
5	0.57	0.57	0.57	SP	5
6	2.17	#N/A	2.17	SB	-2.5
7	4.16	4.16	4.16	SL	-7
8	4.18	4.18	#N/A		
9	5.31	5.31	#N/A	RS	7.5
10	2.60	2.60	#N/A	RD	4.5
11	2.96	2.96	#N/A		
12	3.49	3.49	#N/A	R1	**7.5**
13	1.50	1.50	#N/A	R2	**12**
14	-1.51	-1.51	#N/A		
15	-2.43	-2.43	#N/A		
16	2.74	2.74	#N/A		
17	1.37	1.37	#N/A		
18	1.47	1.47	#N/A		
19	4.74	4.74	#N/A		
20	2.93	2.93	#N/A		
21	3.05	3.05	#N/A		
22	1.92	1.92	#N/A		
23	1.99	1.99	#N/A		

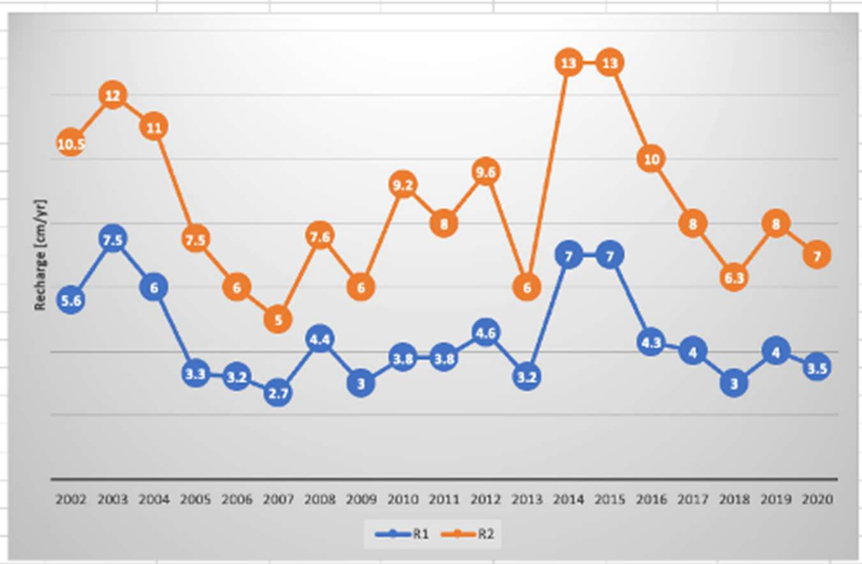
As you examine the plot for each year, you may need to adjust the range of the vertical axis before you can properly fit the lines. To do this, double-click on the vertical axis, click on the axis options tab, and manually adjust the minimum and maximum bounds to properly frame the plot.



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, R2 values, you can see a summary in the Summary sheet.

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., Pulla, S.T., Williams, G.P., Jones, N.L., Mamane, B., Sanchez, J.L., 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., Lo, M., Ho, S.L., Bethune, J., Anderson, K.J., Syed, T.H., Swenson, S.C., de Linage, C.R., Rodell, M., 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., Nelson, J., Swain, N., Christensen, S., Tarboton, D., Dash, P., 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., Pulla, S.T., Jones, N.L., Williams, G.P., David, C.H., Nelson, J.E., Ames, D.P., 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., Famiglietti, J.S., 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., Christensen, S.D., Snow, A.D., Dolder, H., Espinoza-Dávalos, G., Goharian, E., Jones, N.L., Nelson, E.J., Ames, D.P., Burian, S.J., 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, T.H., Famiglietti, J.S., Chambers, D.P., 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., Reigber, Ch., 2002. The Grace Mission: Status And Future Plans 4819. Available at: https://ui.adsabs.harvard.edu/abs/2002EGSGA..27.4819T

Voss, K.A., Famiglietti, J.S., Lo, M., de Linage, C., Rodell, M., Swenson, S.C., 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