
GRACE Groundwater Subsetting Tool

Release 2.0

Norm Jones

Apr 11, 2024

TABLE OF CONTENTS

1	Overview	3
1.1	GGST Web Application	3
1.2	API and Google Colaboratory Notebook	4
2	Computational Algorithm	5
2.1	Derivation of Groundwater Dataset	5
2.2	Grid Subsetting	6
2.3	Uncertainty Estimates	6
2.4	Storage Depletion Curve	7
2.5	Limitations	7
2.6	Software Availability	7
3	Adding and Deleting Regions	9
3.1	Uploading a Region	9
3.2	Deleting a Region	13
4	Analysis at a Single Point	15
4.1	Selecting a Point	15
4.2	Moving or Deleting a Point	17
4.3	Exporting a Time Series	17
5	Application Programming Interface (API)	19
5.1	API Methods	19
5.2	Obtaining an Authentication Token	21
5.3	GGST API Google Colaboratory Notebook	21
6	The Water Table Fluctuation Method	27
6.1	Methods for Estimating Recharge Component	28
6.2	Downloading the Water Level Time Series from the GGST App	29
6.3	Downloading the Water Level Time Series from the API Google Colaboratory Notebook	30
6.4	Gaps in the GRACE Data	32
6.5	Data Imputation Tools	35
6.6	Multi-Linear Trend Analysis	35
6.7	Data Processing Examples	37

The GRACE Groundwater Subsetting Tool (GGST) uses data from the NASA Gravity Recovery And Climate Experiment (GRACE) mission to analyze long-term groundwater storage change for selected regions. GGST can be used to identify and characterize conditions in data-poor areas or identify trends in other regions where trends can be obscured by noise from well data. GGST uses GRACE mission data to compute and display changes in water storage in a web-based mapping system and integrates 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.

These tools were originally developed via funding from the National Aeronautics and Space Administration: 80NSSC20K0155; United States Agency for International Development: Cooperative Agreement with SERVIR West Africa Hub. More details about this project can be found here: <https://hydroinf.groups.et.byu.net/servir-wa/>

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 (J. Famiglietti et al., 2011; J. S. Famiglietti, 2014; Rodell, Velicogna, & Famiglietti, 2009; Thomas, Reager, Famiglietti, & Rodell, 2014).

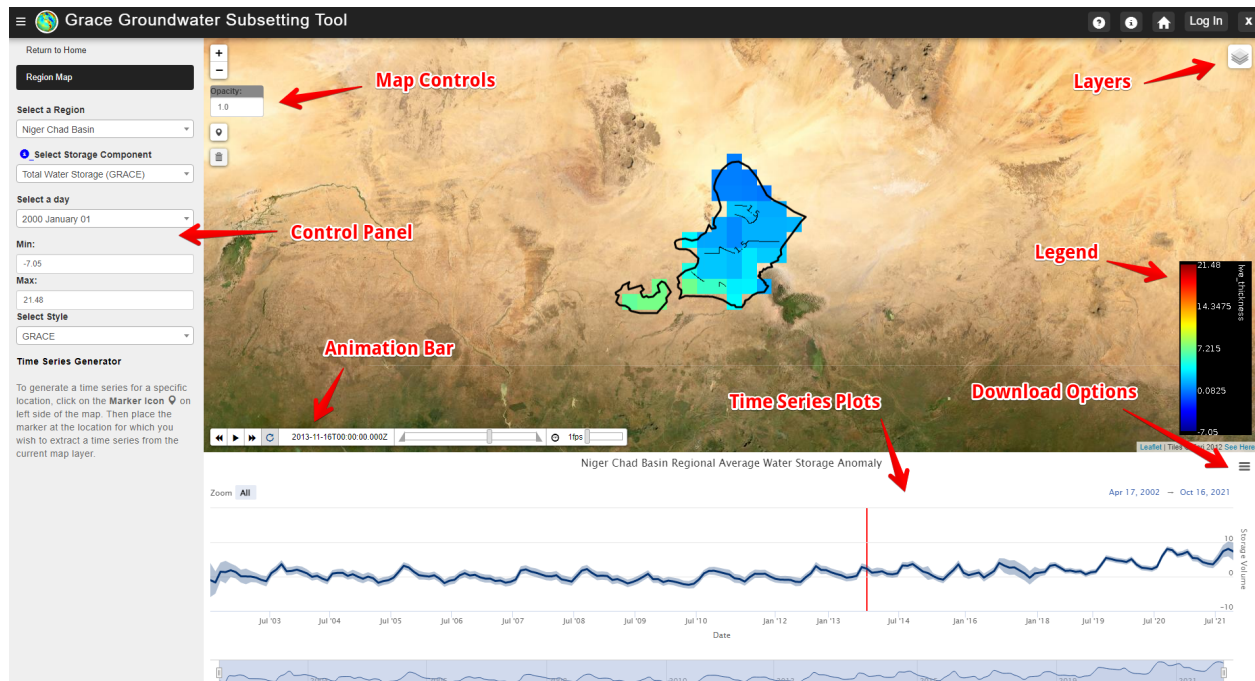
While several tools have been developed for processing and visualizing GRACE data, GGST is designed specifically to support groundwater resource management by regional stakeholders and decision-makers. We accomplish this by carefully processing the raw GRACE data to remove anomalies and improve resolution. 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 on the Algorithm page.

You 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 [Tethys Platform](#). Tethys is a web-based application development framework for rapid deployment of end-user-focused tools that follow modern, consistent, scalable, cross-platform, reusable, web programming paradigms. Tethys is built on commonly used web programming frameworks (e.g., Django, GeoServer, PostGIS, OpenLayers). It is an open-source platform which 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 timeseries 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 you must login with an admin account. If you have administrator (admin) access, follow the instructions on the Adding and Delete Regions page. If not, consider using our API method or feel free to reach out to our 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 all experimental web applications developed by the BYU Hydroinformatics Lab (<https://tethys.byu.edu>).
- **GEOGloWS Portal:** A portal associated with the GEOGloWS organization (<https://apps.geogloss.org/apps/>).

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 designed to run each of the API functions and help you download and visualize the data. For more detailed documentation please visit our API page. Below is a gif produced of the Niger Chad Basin using our Google Colab Notebook. Google Colaboratory is a web based Python programming environment hosted in the Google Cloud that is a component of the Google Drive environment.

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 400km above the Earth and are separated by 200km. 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 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 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 netCDFs 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.

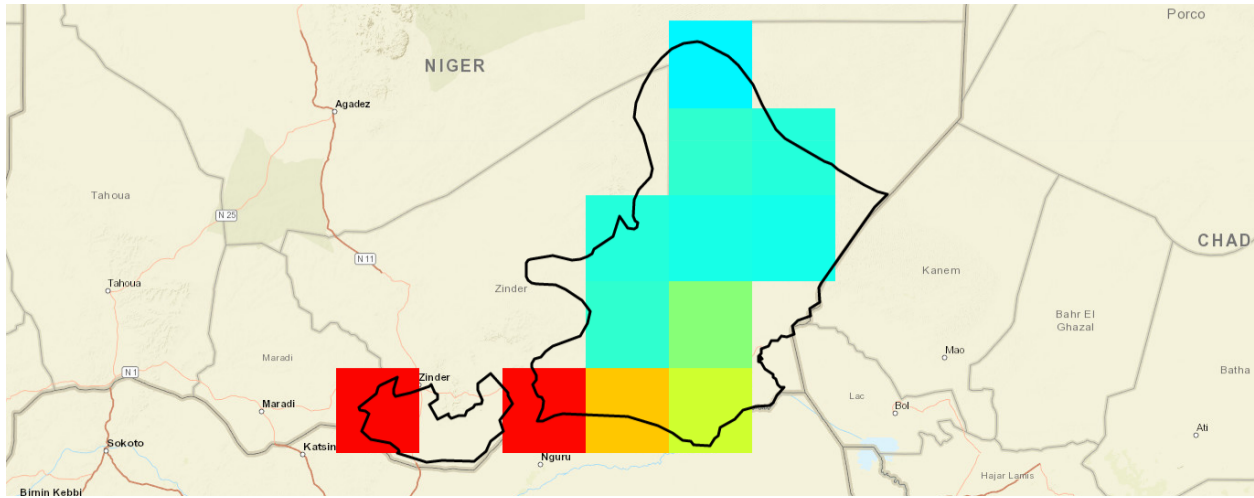
The converted files are then used to compute the groundwater anomaly using a mass balance approach. The groundwater anomaly is the difference between the TWSa and the sum of the surface water component anomalies.

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

The result of this computation is the ground water 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 subsetting 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.

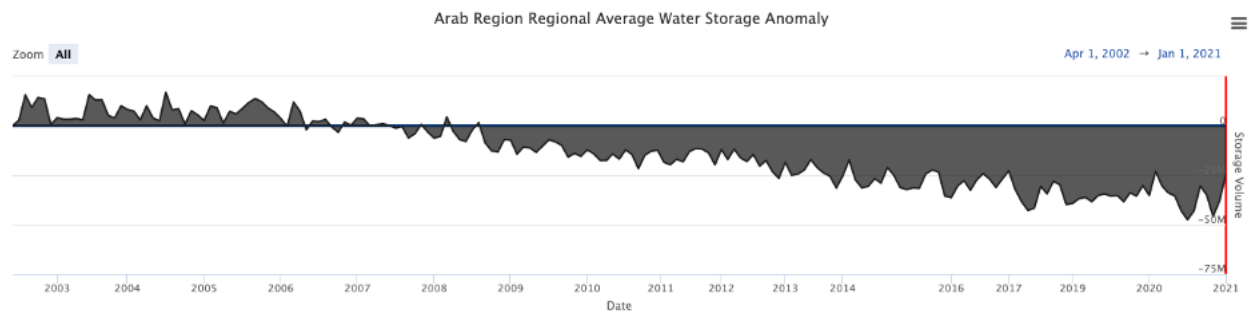
2.4 Storage Depletion Curve

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. Raw GRACE data is at an even coarser resolution (3-degrees latitude by 3-degrees longitude) which is then processed to higher resolutions TWSa data.

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.

It is also recommended 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. It can be accessed on a Tethys portal associated with the NOAA GeoGLOWS project by browsing to this [link](#) and selecting the Grace Groundwater Subsetting Tool application.

ADDING AND DELETING REGIONS

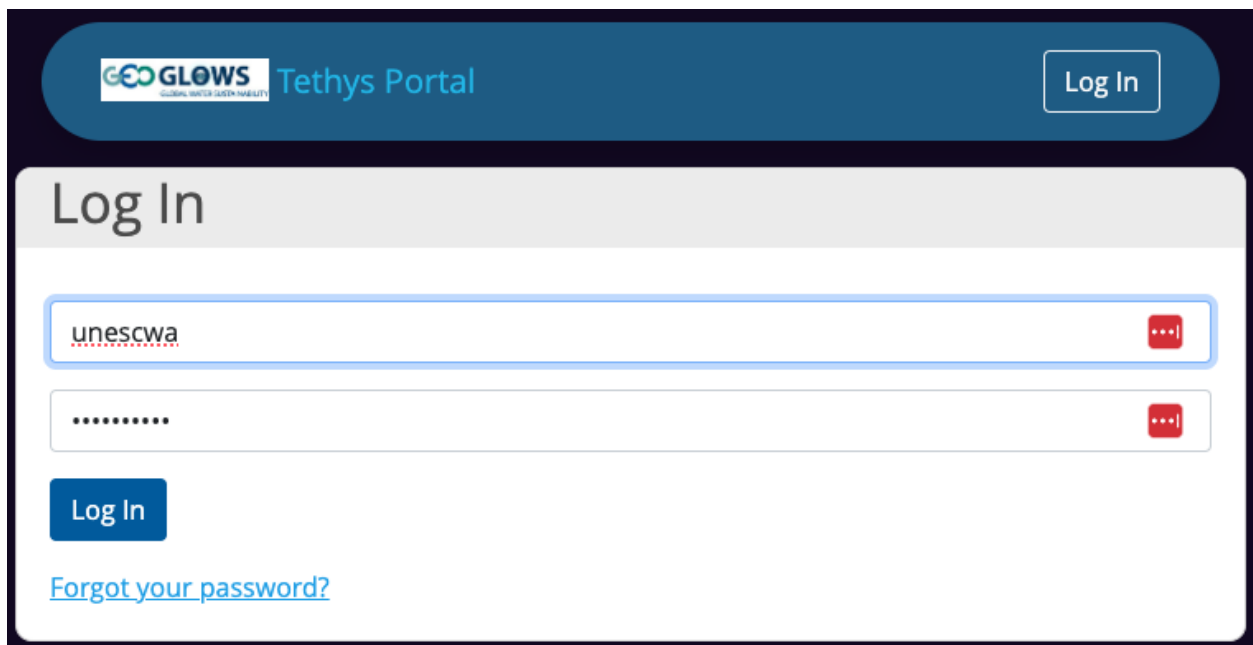
This section describes how to upload new regions to the GGST app. When a new region is uploaded, it is automatically processed and the storage components including subsetting netCDF files and storage time series are computed for the region and stored with the region for visualization in the app. The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

3.1 Uploading a Region

The GRACE Groundwater Subsetting Tool web application is hosted on three different Tethys portals:

- **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 Main Portal:** A Tethys portal hosted by Brigham Young University for all completed apps (<https://tethys.byu.edu>).
- **BYU Staging Portal:** A Tethys portal hosted by Brigham Young University for testing new applications or features (<https://tethys-staging.byu.edu/apps/>).

To upload regions on the application, visit the portal of your choice and log in using the “Log In” link in the upper right corner of the portal window.

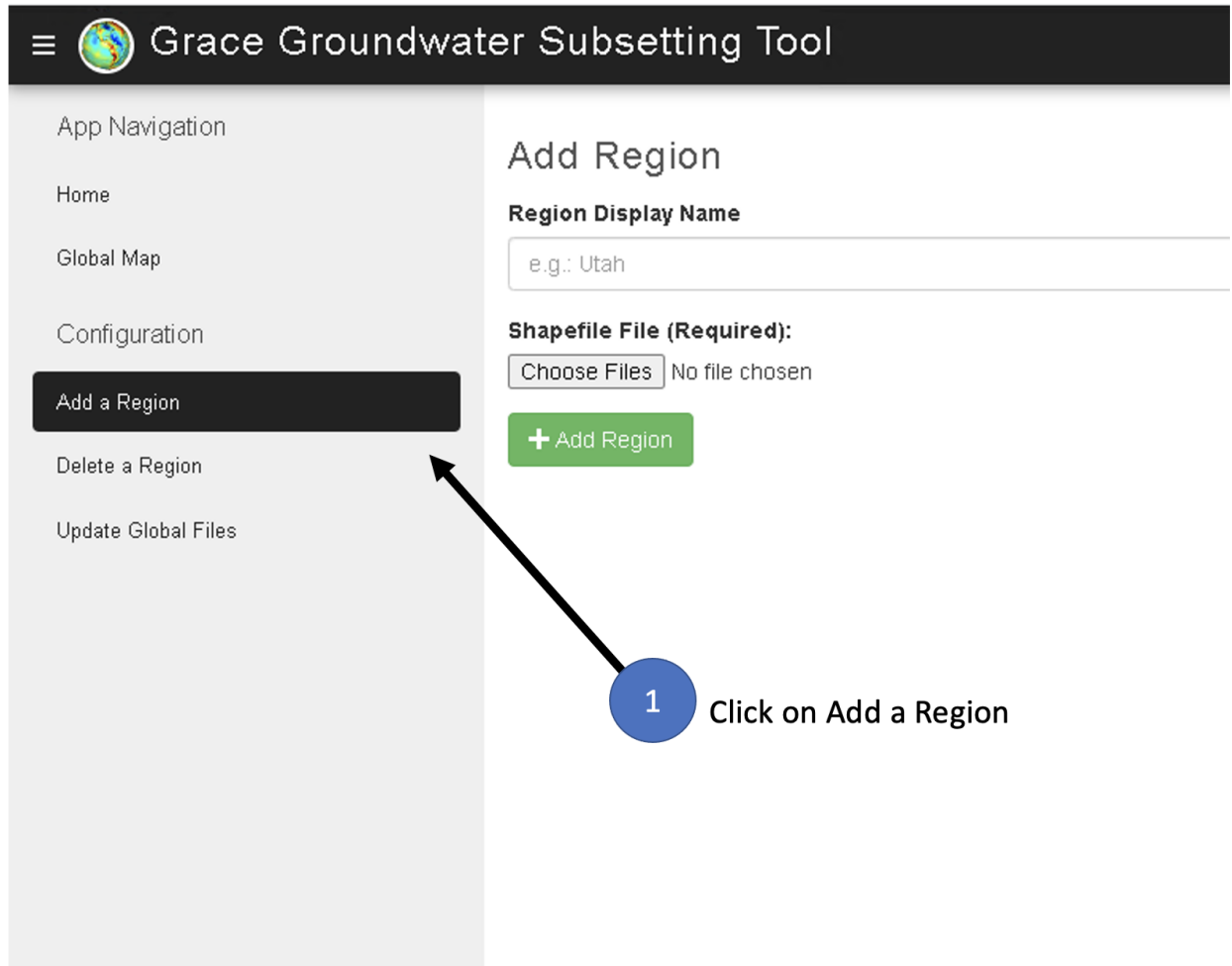


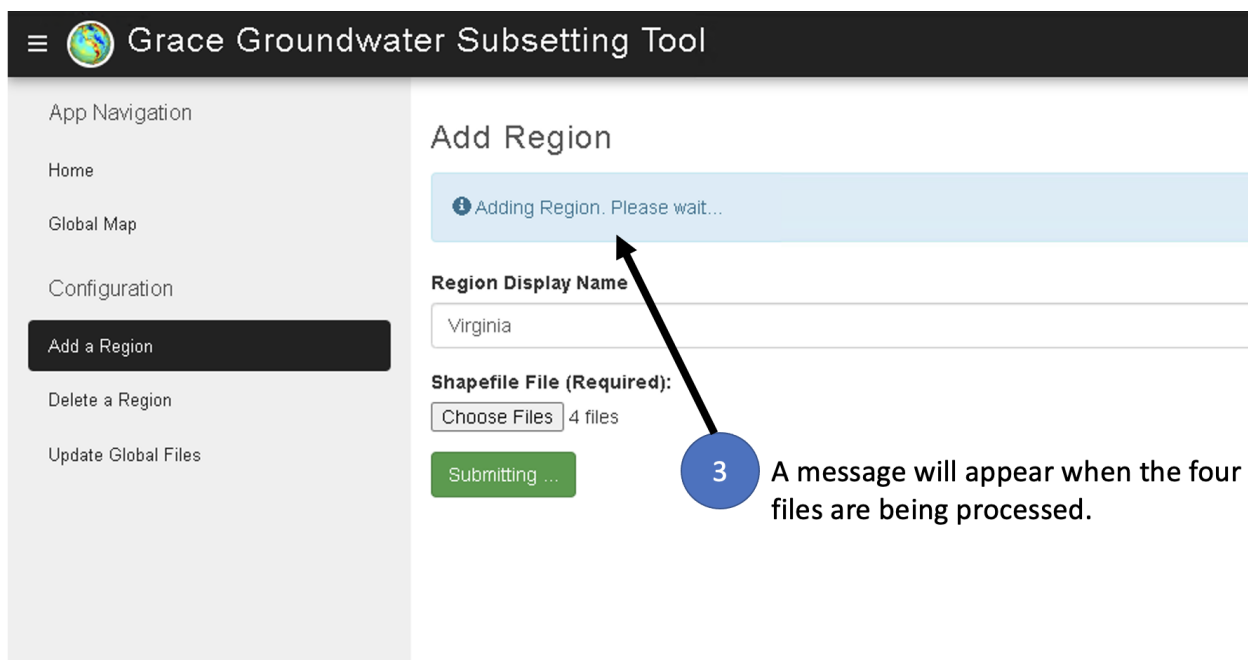
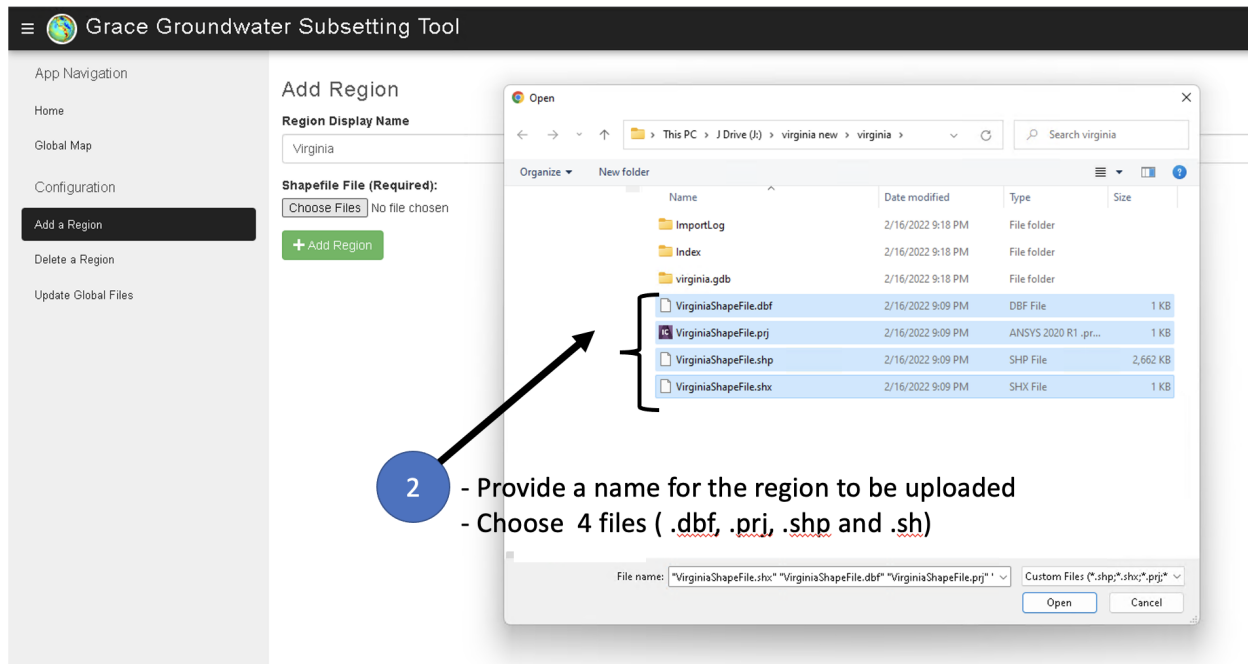
The screenshot displays the login page of a Tethys Portal. At the top, a dark blue header bar contains the GEOGLWS logo on the left, the text "Tethys Portal" in the center, and a "Log In" button on the right. Below the header, the main content area has a light gray background. It features a "Log In" heading, followed by a username input field with the text "unescwa" and a red eye icon to toggle visibility. Below the username field is a password input field with masked characters "....." and a red eye icon. A blue "Log In" button is positioned below the password field. At the bottom of the login area, there is a blue link that says "Forgot your password?".

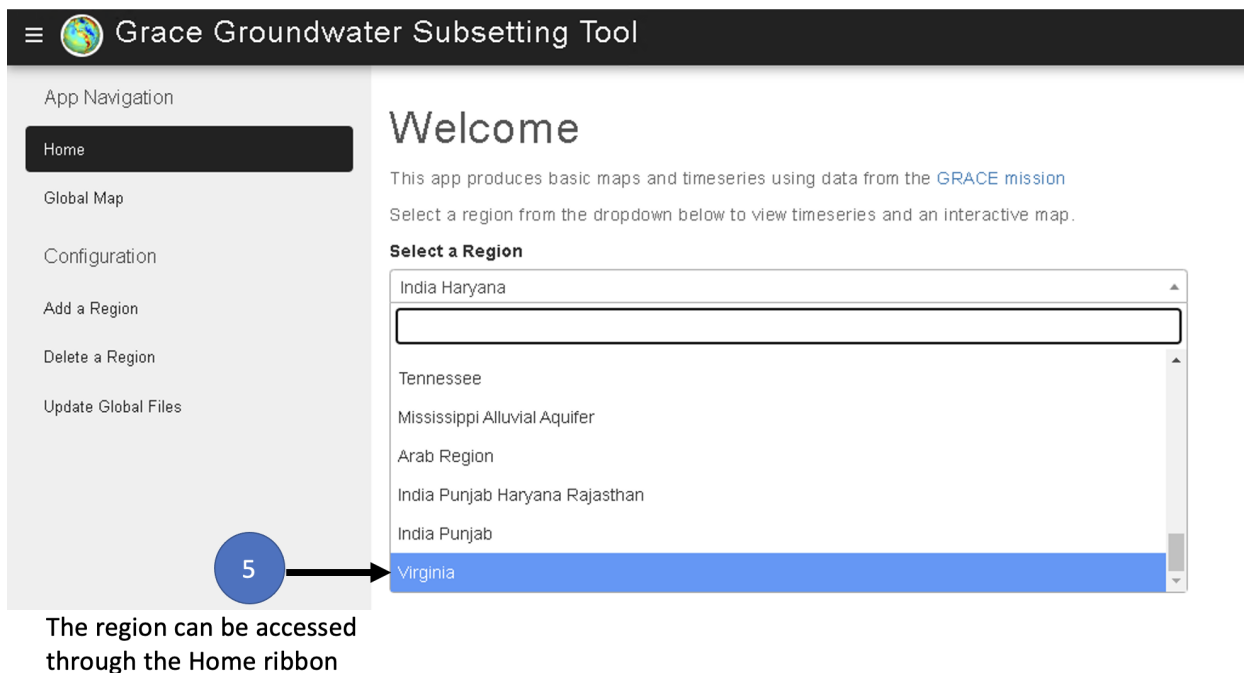
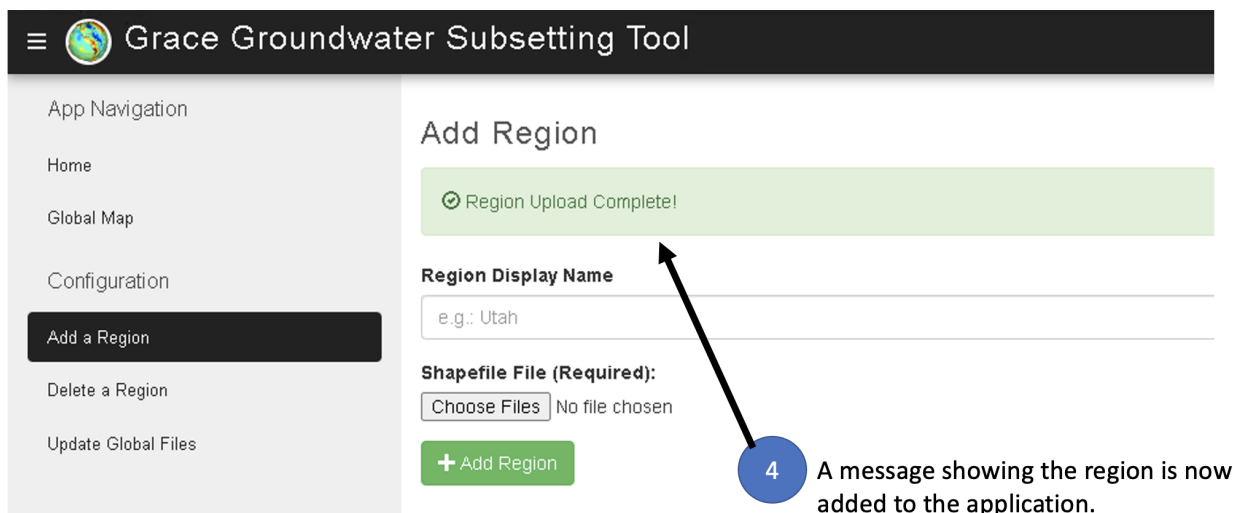
Without logging in you can see the App Navigation pages: Home and Global Map. These allow you to view previously uploaded regions and create time series graphics for any singular point on the globe. Once you log in with administrative privileges, you will see the additional Configuration pages: Add a 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:







When uploading a region, it is recommended that the region be at least 3x3 degrees in size. Smaller regions can be processed, but the uncertainty in the results increases. This is because the native GRACE grid cells are 3x3 degrees in resolution before downscaling to 0.5x0.5 degrees. The GLDAS grid cells are 1x1 degree and therefore the resulting Groundwater Storage Anomaly (GWSa) cells are 1x1 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 following error message is displayed:

Add Region

i Info: Adding Region. Please wait...

✕ *Error: Error processing request: No data found in bounds. Data variable: lwe_thickness


Region Display Name

Palestine East

Shapefile File (Required): 4 files

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.

≡  Grace Groundwater Subsetting Tool

App Navigation

Home

Global Map

Configuration

Add a Region

Delete a Region

Update Global Files

Delete Region

Select a Region

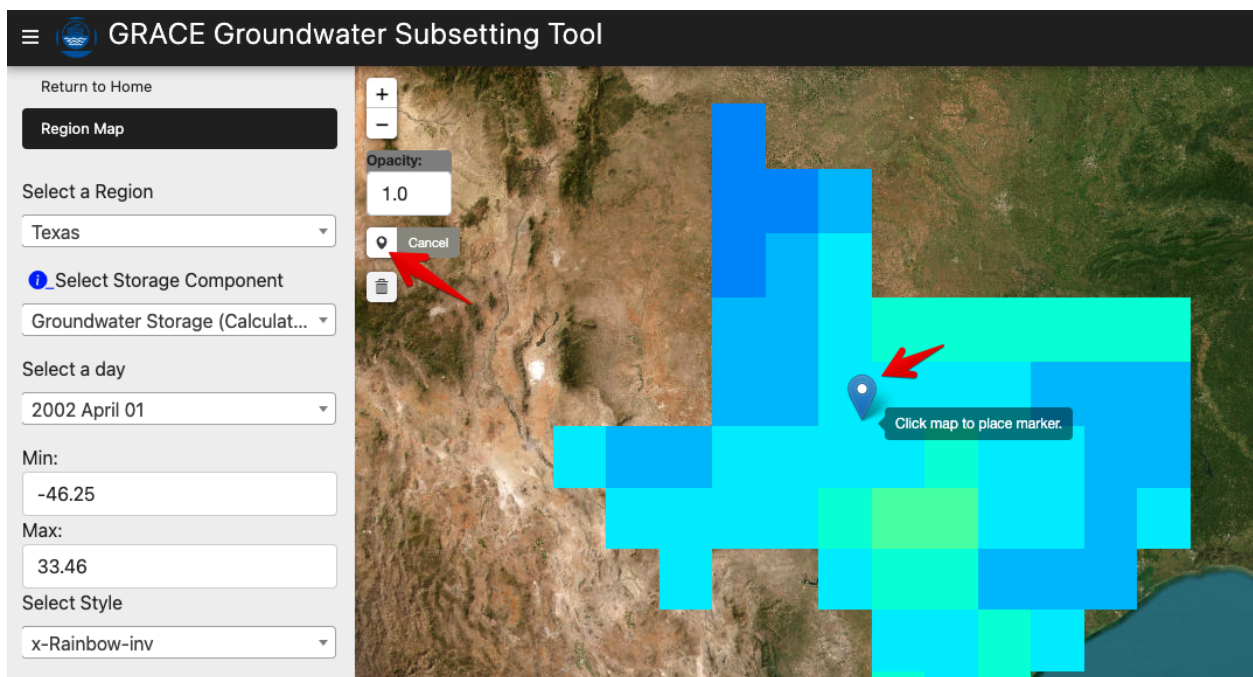
Niger Chad

ANALYSIS AT A SINGLE POINT

In addition to analyzing groundwater storage change averaged over a region, the GGST app can be used to perform an analysis at a single point location. This can be used to quickly generate a time series at a point of interest or in cases where a region of interest is too small to be processed using the Upload Region feature. For a point analysis, GGST finds the GRACE and GLDAS grid cells containing the selected point and returns the selected dataset time series for the cell.

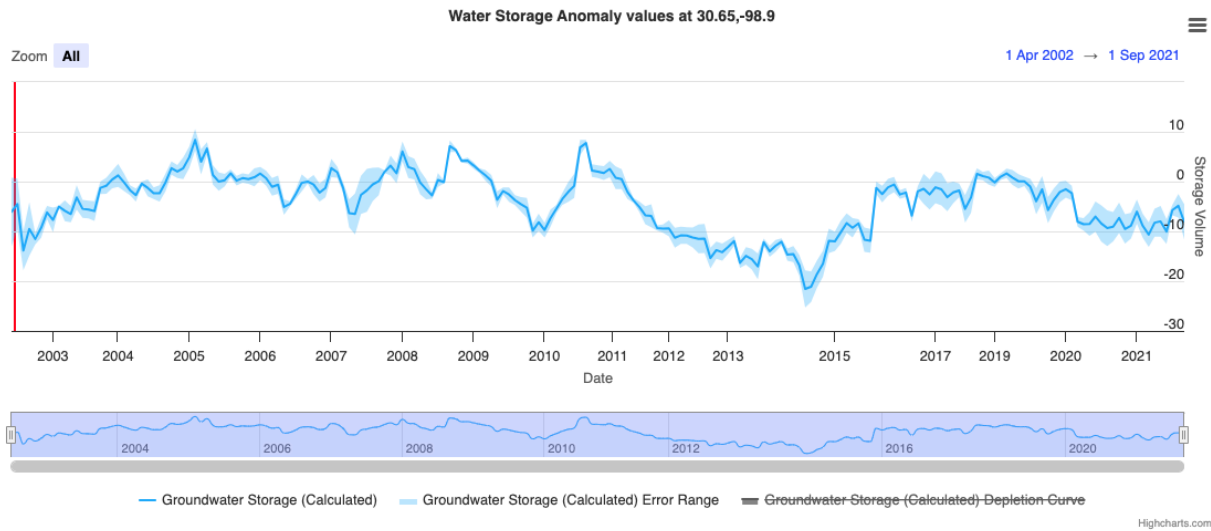
4.1 Selecting a Point

Before performing a point analysis, you should first zoom in to your area of interest. You can do this using either the global view, or by selecting a region that has previously been uploaded. You should then select the desired storage component using the “Select Storage Component” option. These options will include: Total Water Storage (GRACE), Surface Water Storage (GLDAS), Soil Moisture Storage (GLDAS), and Groundwater Storage (Calculated). After selecting a component, click on the Select Point tool and then click on the map where you wish to perform a point analysis. If you are viewing a region, the selected point must be within the bounds of the region.



After clicking on a point, a time series plot for the selected storage component at the point location appears:

Point Timeseries



Close

If the Time Series window is closed, it can be relaunched using the “View Time Series” button on the left.

default

Time Series Generator

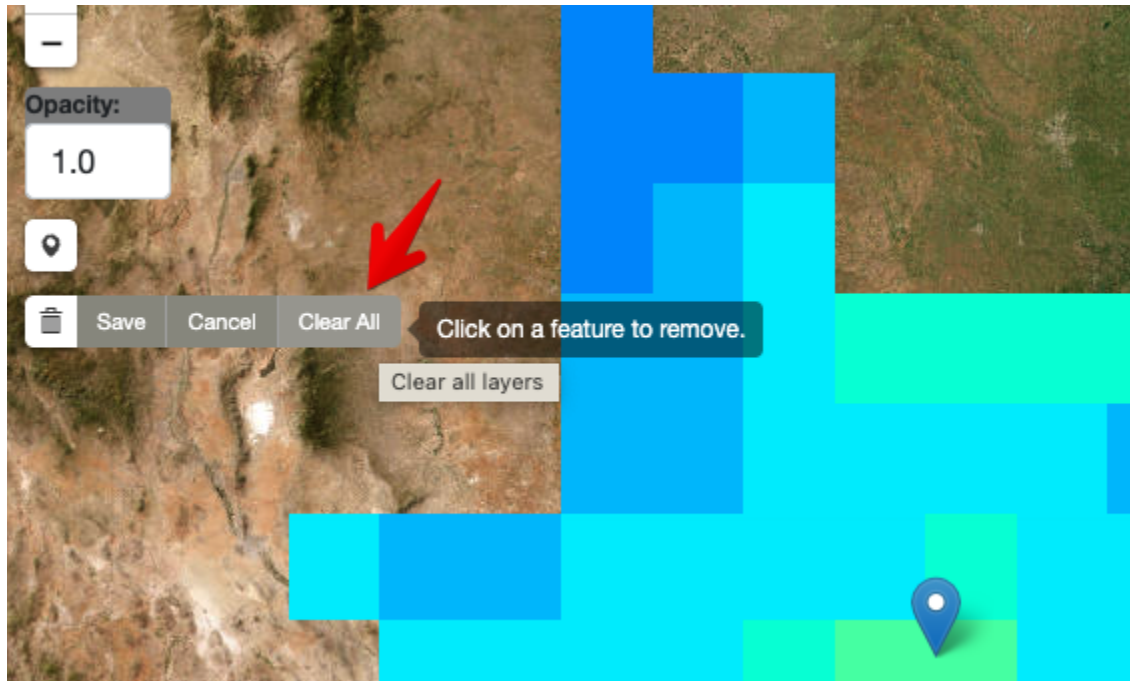
View Point Timeseries

To generate a time series for a specific location, click on the **Marker Icon** on left side of the map. Then place the marker at the location for which you wish to extract a time series from the current map layer.

To perform a time series analysis at a new point, close the Time Series window and click on a new point while the Select Point tool is active.

4.2 Moving or Deleting a Point

To delete an existing point, click on the Delete tool and select the “Clear All” option.



4.3 Exporting a Time Series

To export the data associated with a point time series, click on the menu icon in the upper right corner of the Time Series window and export to the selected format:



When exporting a time series, the data in the time series are saved to a tabular file with columns for the date, and each component of the time series is exported as a separate column. The date column is saved in a unique format representing the number of milliseconds since January 1, 1970. This can be converted to a more typical date format using a spreadsheet formula. This process is described in the [Water Table Fluctuation Method](#) section.

APPLICATION PROGRAMMING INTERFACE (API)

The Python API for the GGST allows users to retrieve ground water 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 section 4.2 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. An example implementation using the Python code language in a Google Colaboratory notebook has been provided below. Before using the API in the Google Colaboratory notebook, each of the four GGST API methods will be explored in the following section.

5.1 API Methods

All four GGST 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¶m2=value2&...paramN=valueN](https://tethys-staging.byu.edu/apps/[parent-app]/api/[MethodName]/?param1=value1¶m2=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`

Let's explore each API method individually and offer an example:

1. The `getStorageOptions` Method

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` Method

Parent Application	GGST			
Supported Methods	GET			
Returns	A JSON object with a timeseries 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 (-60,180)	Yes
	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

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

https://tethys-staging.byu.edu/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 Colab Notebook for further instructions. See details below on how to obtain one.

3. The getRegionTimeseries Method

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

Parent application	GGST			
Supported Methods	POST			
Returns	A zip file with regional netCDF files for each storage option clipped to the uploaded shapefile.			
Parameters(Params)	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 no files.

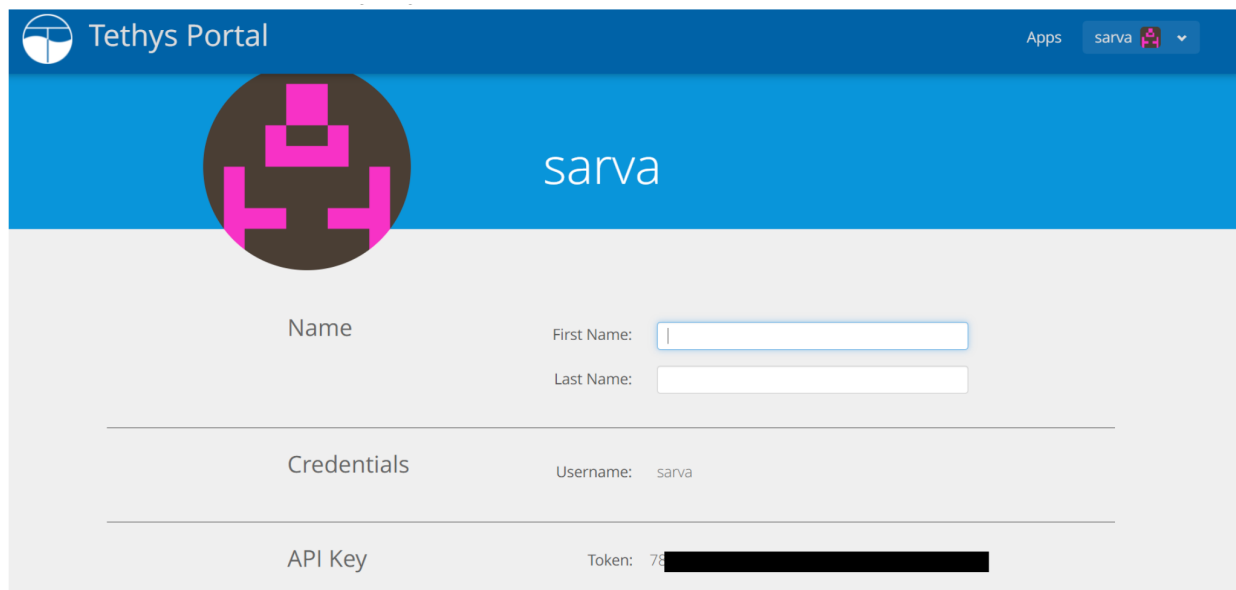
5.2 Obtaining an Authentication Token

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](#). 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.

5.3 GGST API Google Colaboratory Notebook

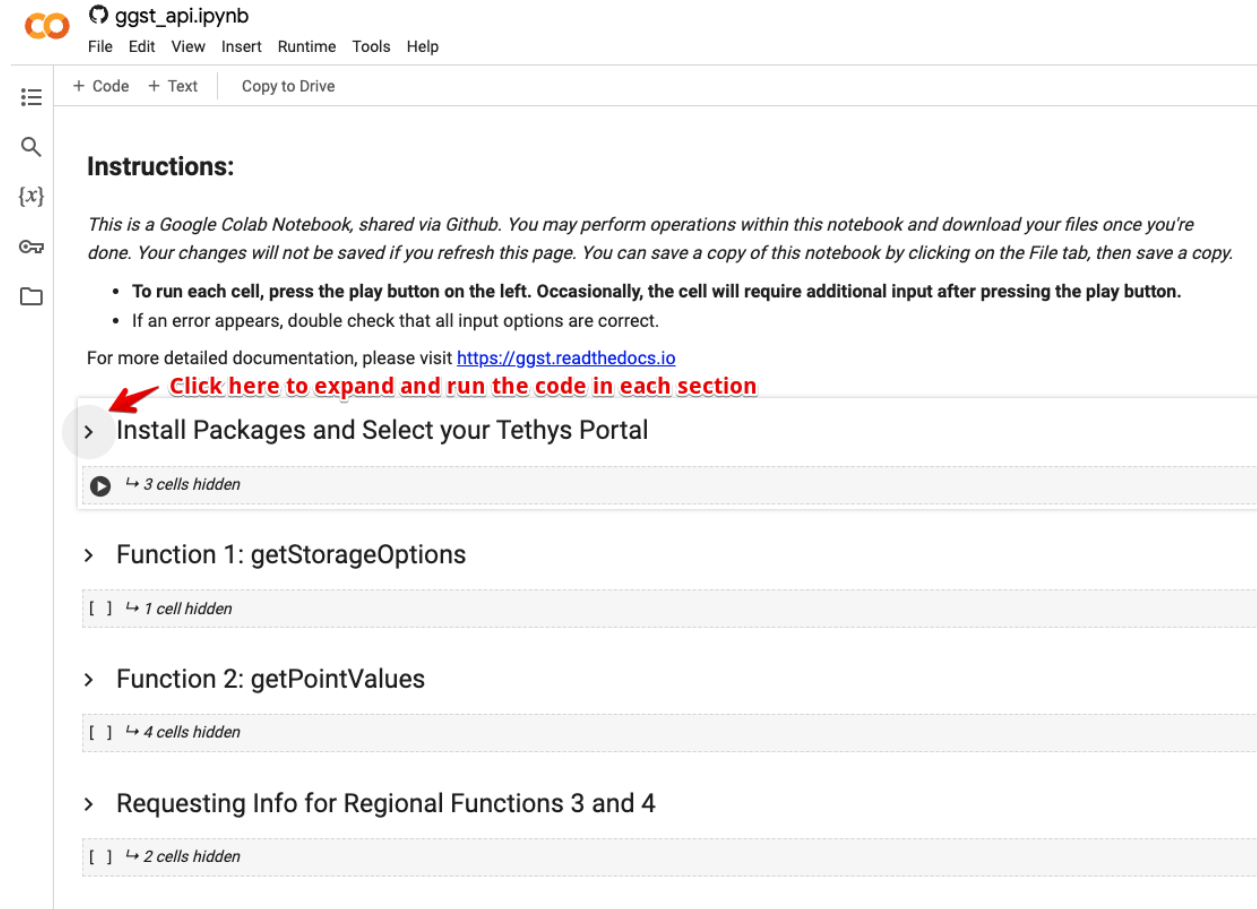
We have provided an example of calling the GGST API using the Python coding language in a Google Colab Notebook. You can open the notebook with this link:

You may wish to make a copy of the notebook in your own Google Drive.

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 notebook is divided into multiple sections and each section contains a set of cells, each of which contains 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:



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

Function 2: getPointValues

```
[ ] 1 #@markdown ## **Set desired inputs for getPointValues function and run this
2 Latitude = 31.2483#@param {type:"number"}
3 Lat = str(Latitude)
4 Longitude = 34.6762#@param {type:"number"}
5 Lon = str(Longitude)
6 F2_Storage_Option = "gw" #@param ["gw", "sm", "sw", "swe", "tw", "canopy",
7
8 #Run getPointValues function with your inputs
9 pointvalues = requests.get(portal + "/apps/ggst/api/getPointValues/?latitude
```

Set desired inputs for getPointValues function and run this cell

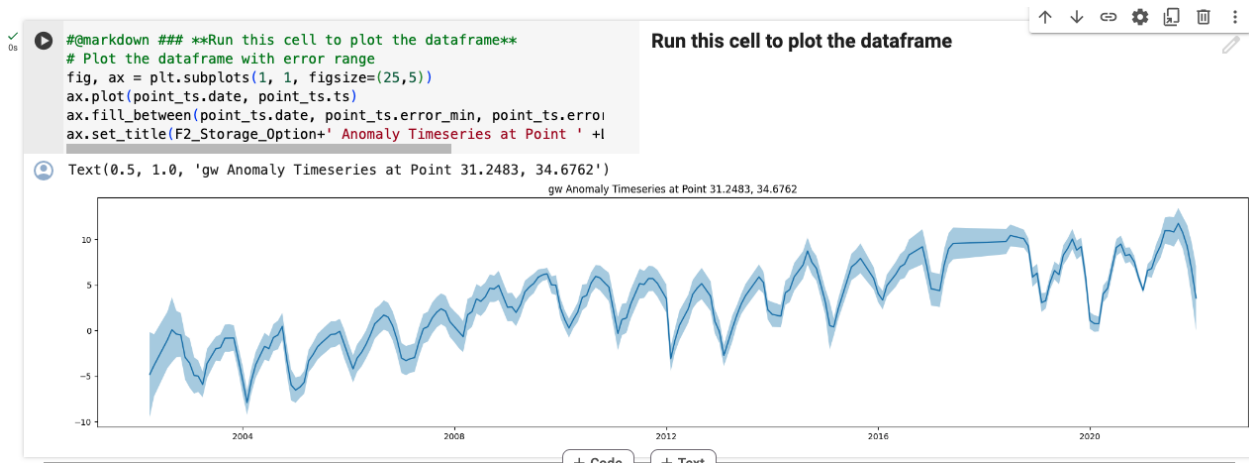
Latitude:

Longitude:

F2_Storage_Option:

```
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
```

5.3. GGST API Google Colaboratory Notebook



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.

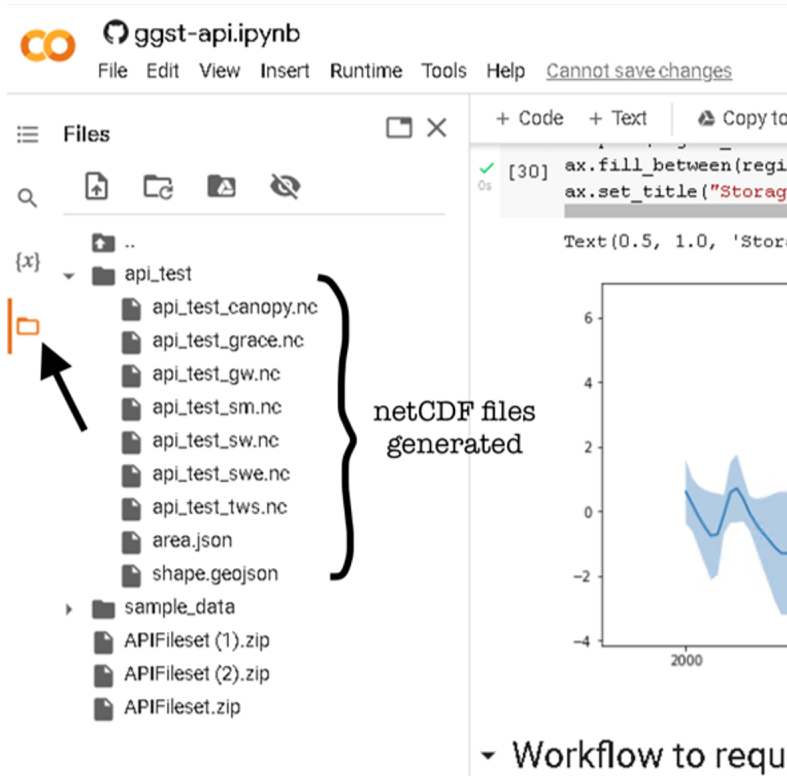
- **Function 3: getRegionTimeseries**

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:

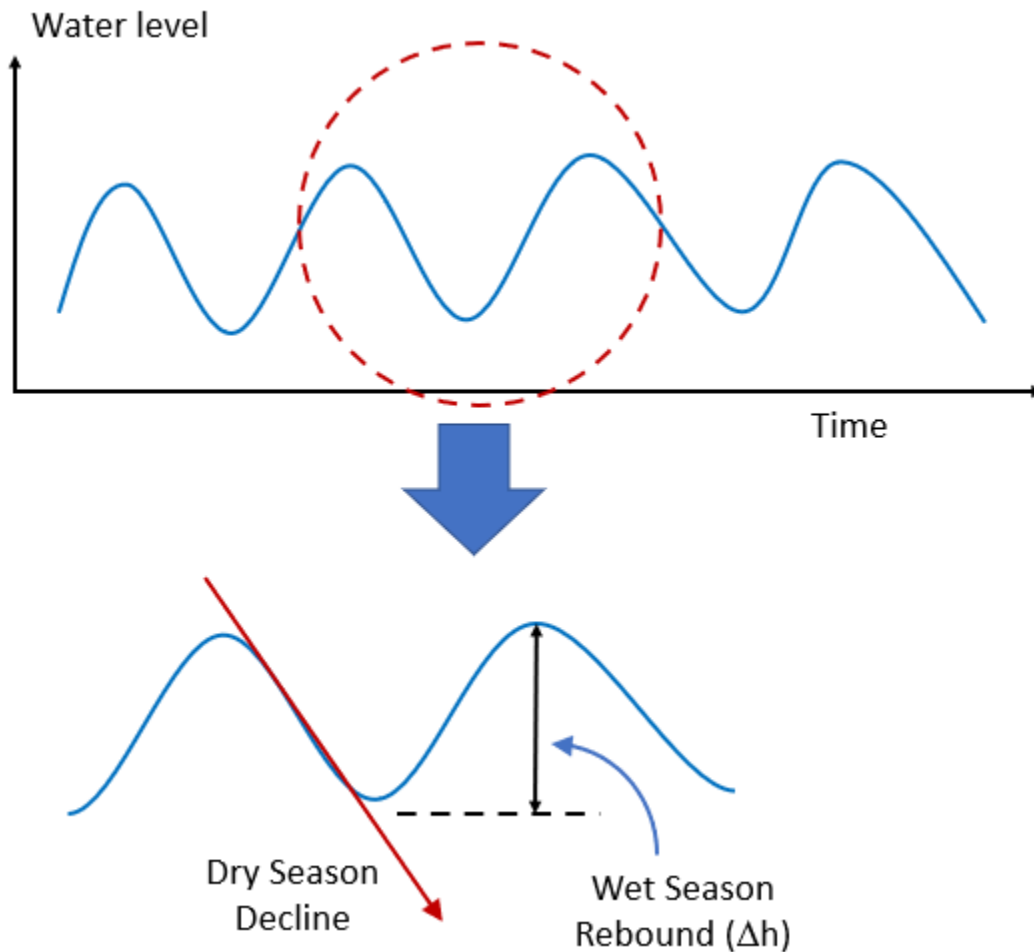
Name	Abbreviation	Source	Source Resolution
Total Water Storage	grace	GRACE	0.5 degrees
Surface Water Storage	sw	GLADAS	1.0 degrees
Soil Moisture Storage	sm	GLADAS	1.0 degrees
Groundwater Storage	gw	Calculated*	1.0 degrees
Snow Water Equivalent	swe	GLADAS	1.0 degrees
Terrestrial Water Storage	tws	GLADAS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

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 S_y 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.

6.1 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:

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

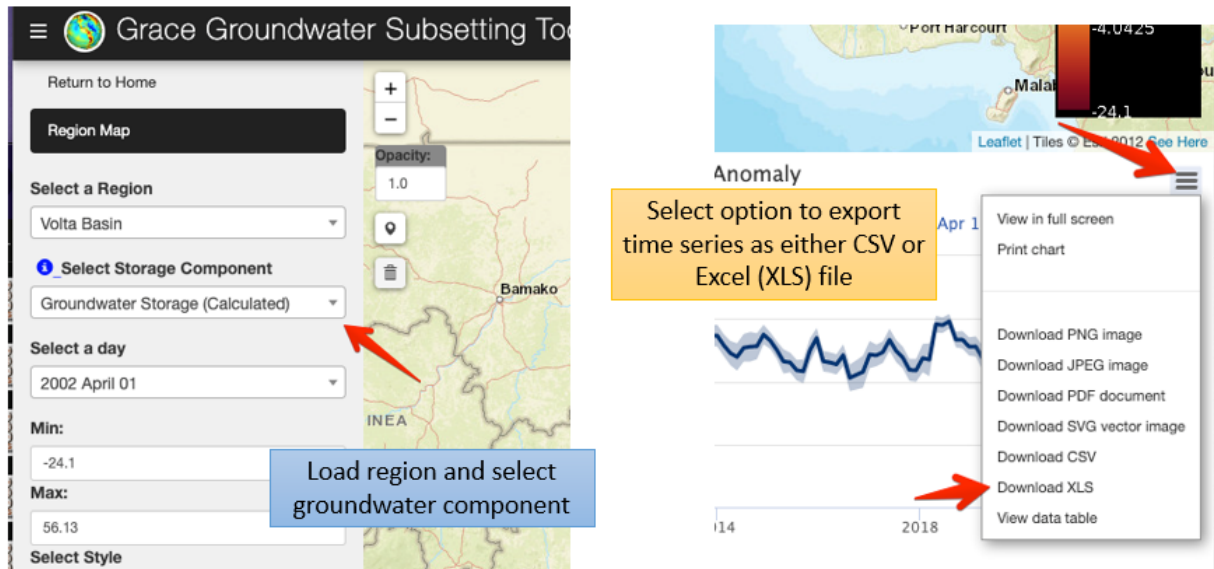
Another method is to assume that the groundwater decline as a result of pumping and discharge continues at the same rate in the wet season and therefore the rise should be computed from a linear extrapolation of the declining line as follows:

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

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.](#)

6.2 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
	Date	Groundwater Storage (Calculated)	Groundwater Storage (Calculated) Error Range (low)	Groundwater Storage (Calculated) Error Range (high)
1				
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	-9.07	-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

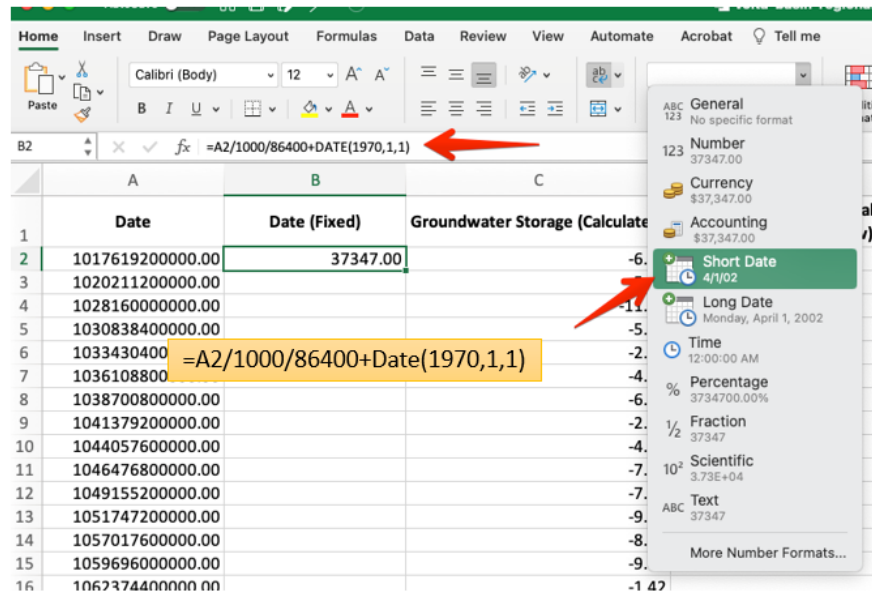
The storage units are liquid water equivalent in cm, as expected, but the date units are reported in milliseconds since Jan 1, 1970. In order to convert to a more typical date unit, we must first create a new column and then enter the formula

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 the one of the standard date options. Whether it appears as month/day/year or day/month/year will depend on your regional settings.

- 1) Create new column
- 2) Enter formula
- 3) Change to date format
- 4) Copy down

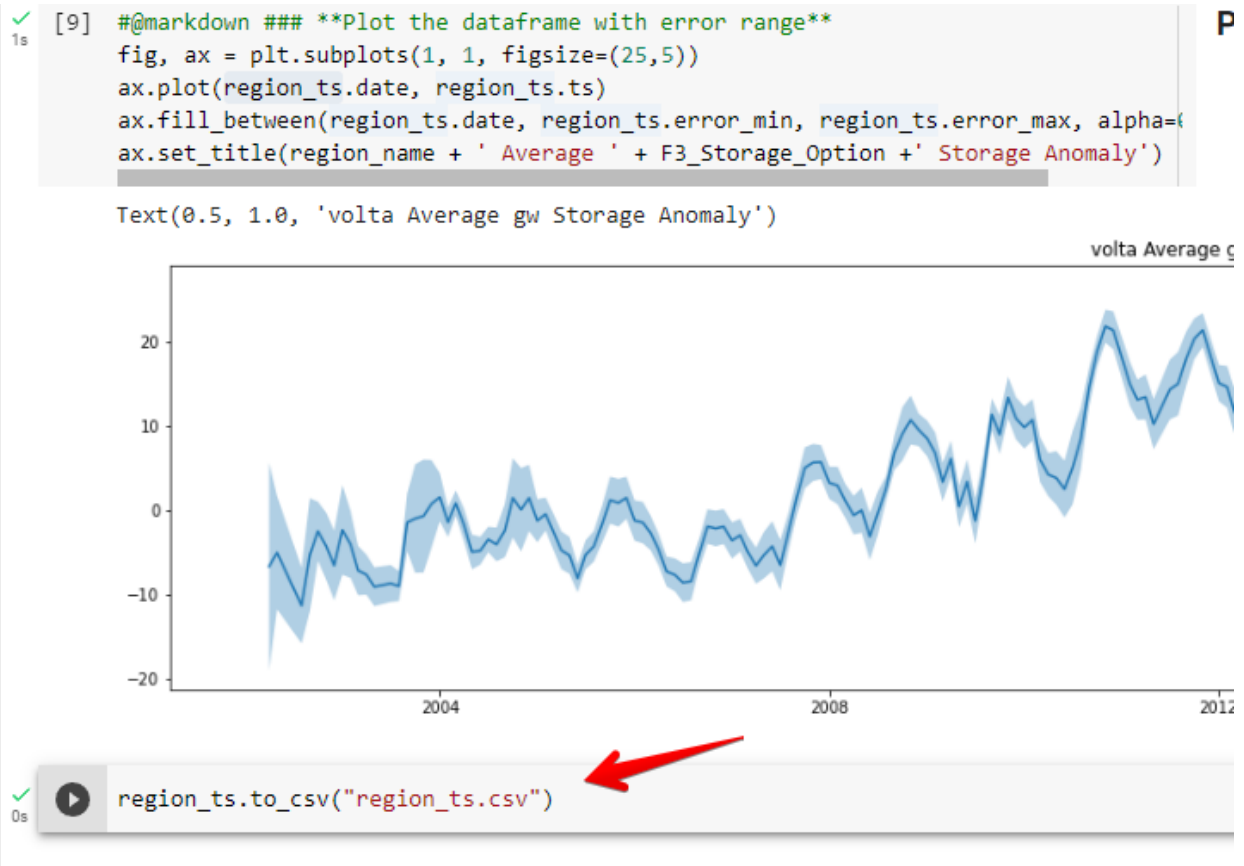
Result:

	A	B	
	Date	Date (Fixed)	Groundw
2	101761920000.00	4/1/02	
3	102021120000.00	5/1/02	
4	102816000000.00	8/1/02	
5	103083840000.00	9/1/02	
6	103343040000.00	10/1/02	
7	103610880000.00	11/1/02	
8	103870080000.00	12/1/02	
9	104137920000.00	1/1/03	
10	104405760000.00	2/1/03	



6.3 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 Colab interface on the left. Click the three vertical dots to the right of the file and select the Download option.

Files

sample_data

region_ts.csv

volta_basin3.zip

+ Code + Text Copy to

✓ [8] 205 2022-04-01 18.9

206 2022-05-01 16.0

207 2022-06-01 15.0

208 2022-07-01 16.9

Download

Rename file

Delete file

Copy path

Refresh

Text(0.5, 1.0, 'volt

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

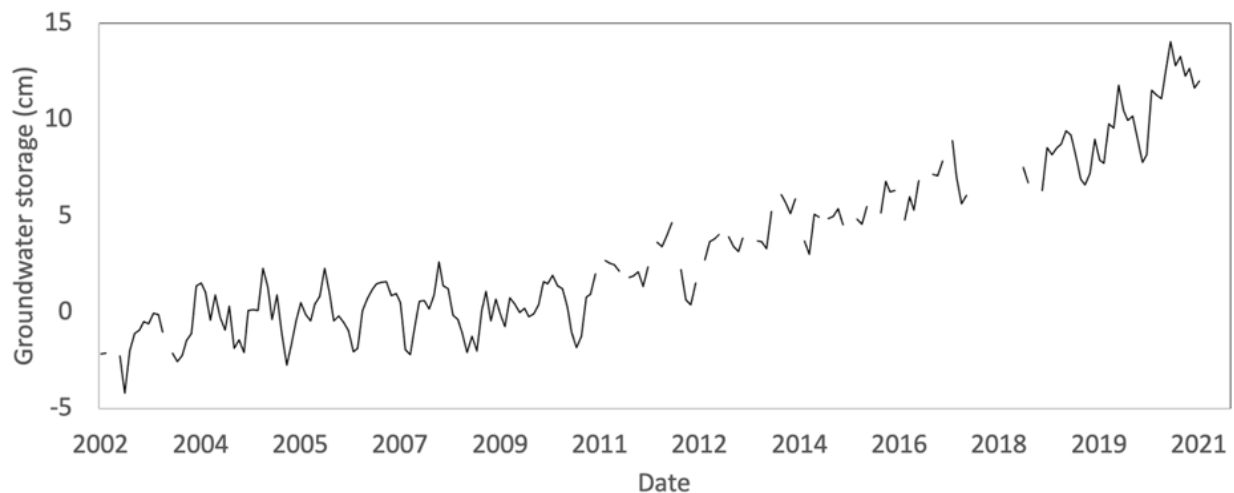
	A	B	C	D	E	F
1		date	ts	error_min	error_max	
2	0	4/1/2002	-6.66	-18.978	5.658	
3	1	5/1/2002	-5.033	-11.749	1.683	
4	2	8/1/2002	-11.323	-15.81	-6.835	
5	3	9/1/2002	-5.273	-11.971	1.426	
6	4	10/1/2002	-2.464	-5.961	1.034	
7	5	11/1/2002	-4.24	-8.214	-0.267	
8	6	12/1/2002	-6.551	-10.684	-2.419	
9	7	1/1/2003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

6.4 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	-1.063	-11.364	-6.762	
14	12	7/1/2003	-3.686	-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	-1.138	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:

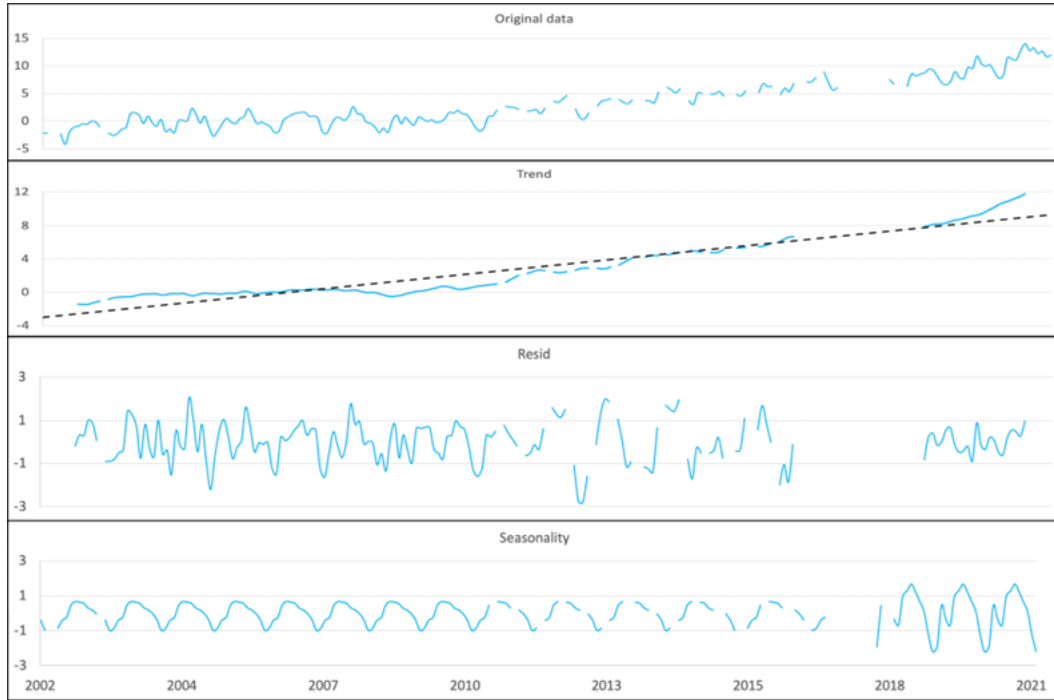


For the years with large gaps, it can be difficult to identify seasonal trends and apply the WTF method. One way to resolve this problem is to use a statistical algorithm to detect seasonal patterns in the data and impute synthetic data in the gaps. This can be accomplished using a simple seasonal decomposition 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:

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

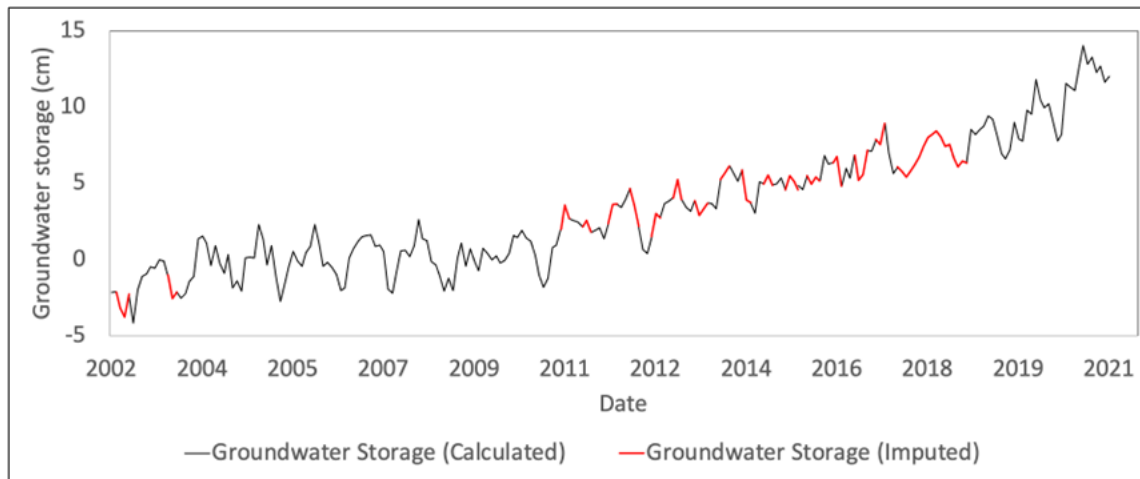
Where $Y[t]$ is the GWSa, $T[t]$ is the GWSa trend, $S[t]$ is the seasonal GWSa component, and $e[t]$ is the residual GWSa component. The decomposition components for the data shown above are as illustrated here:



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:

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


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



6.5 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

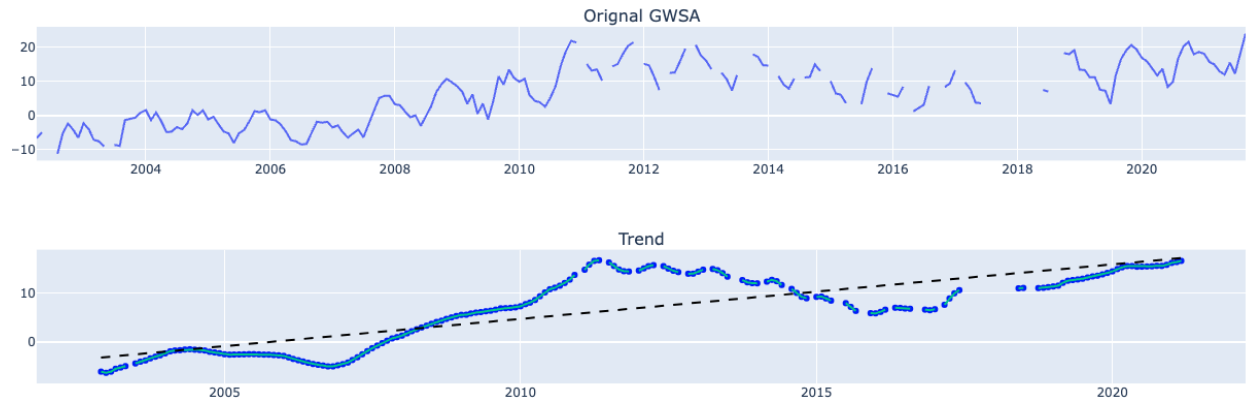
	A	B	C
1	date	ts	
2	4/1/2002	-6.66	
3	5/1/2002	-5.033	
4	8/1/2002	-11.323	
5	9/1/2002	-5.273	
6	10/1/2002	-2.464	
7	11/1/2002	-4.24	
8	12/1/2002	-6.551	
9	1/1/2003	-2.345	
10	2/1/2003	-4.075	
11	3/1/2003	-7.156	
12	4/1/2003	-7.638	
13	5/1/2003	-9.063	
14	7/1/2003	-8.686	
15	8/1/2003	-9.002	
16	9/1/2003	-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	

At this point, the file is ready to be used with the Colab notebook. The following file is an example of a file prepared in the manner described above:

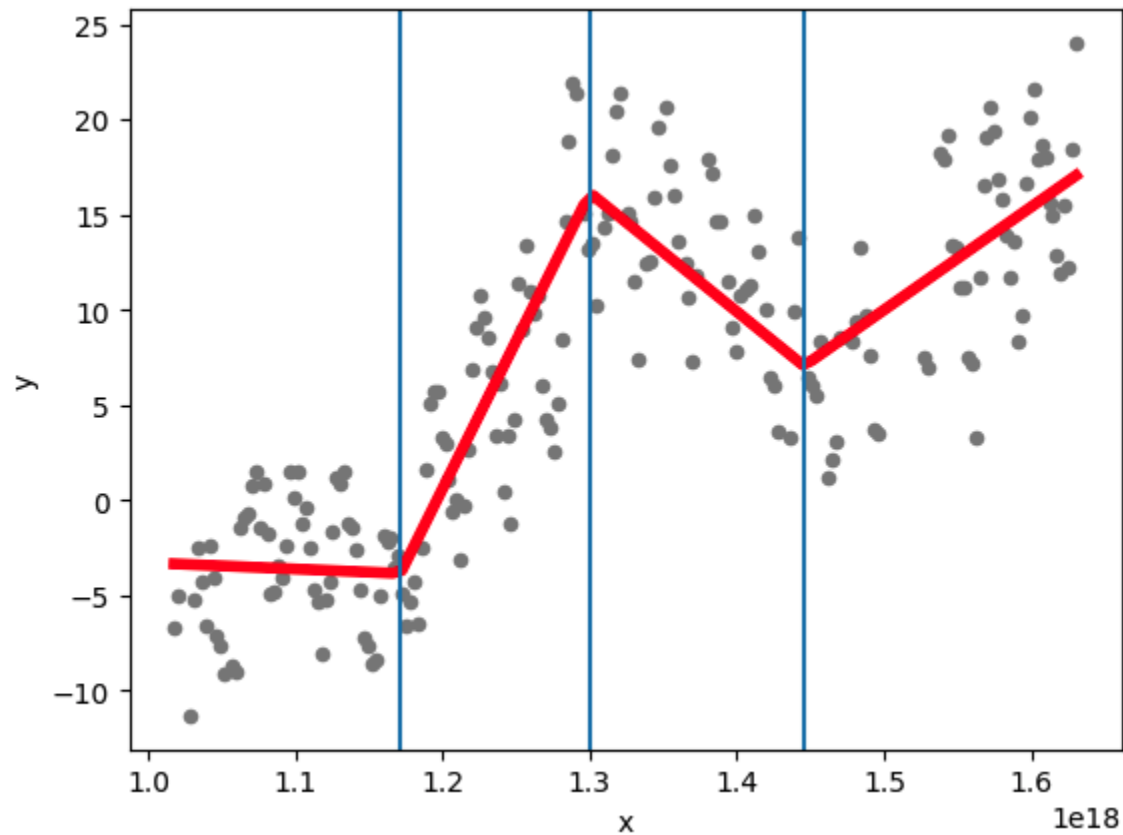
Here is a sample file you can use with the script: `west-gwsa-raw-clean.csv`

6.6 Multi-Linear Trend Analysis

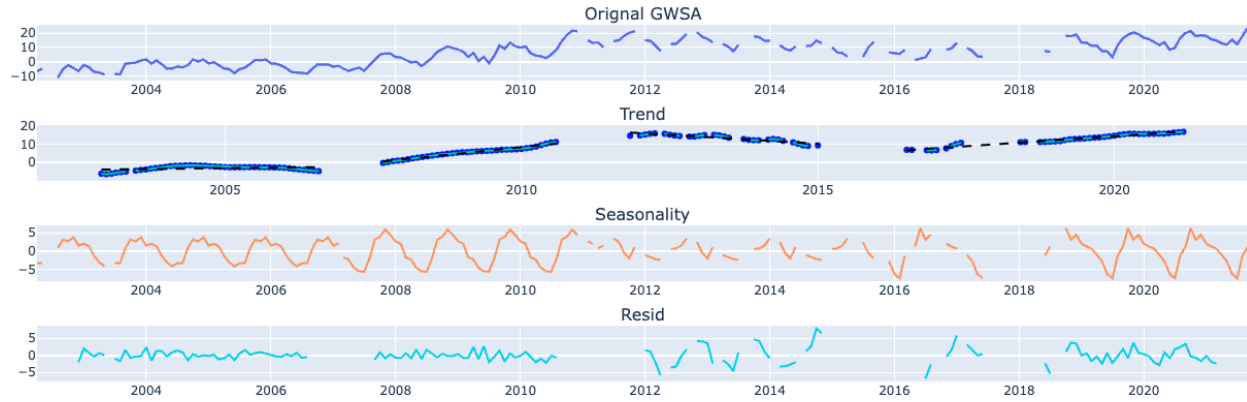
In the seasonal decomposition method described above for gap imputation, a single linear trend was described. Here is the trend resulting from the sample file linked above with a single trend line:



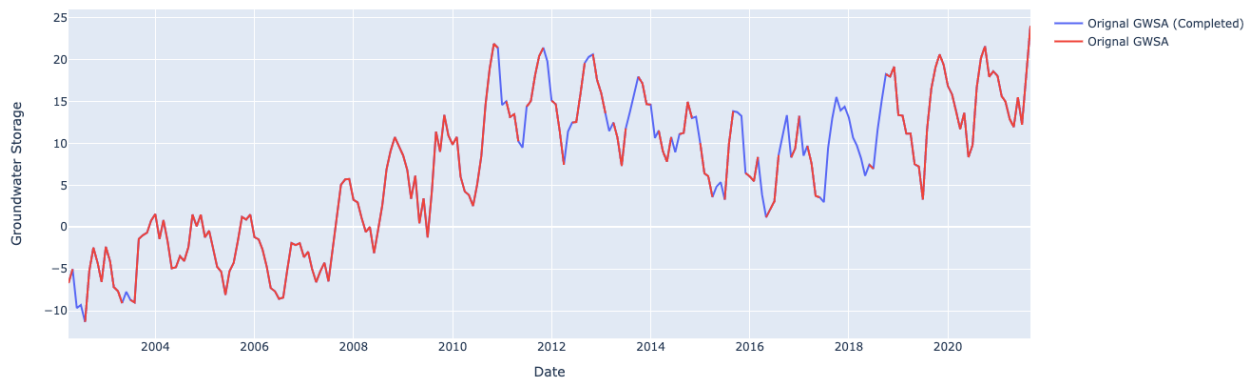
However, many data sets exhibit multiple linear trends. For this dataset, there are four distinct trends. The Python script has an option to perform a multi-linear regression analysis. For this dataset, we set the `number_breakpoints` variable to 3, and run a multi-linear regression algorithm that fits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



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

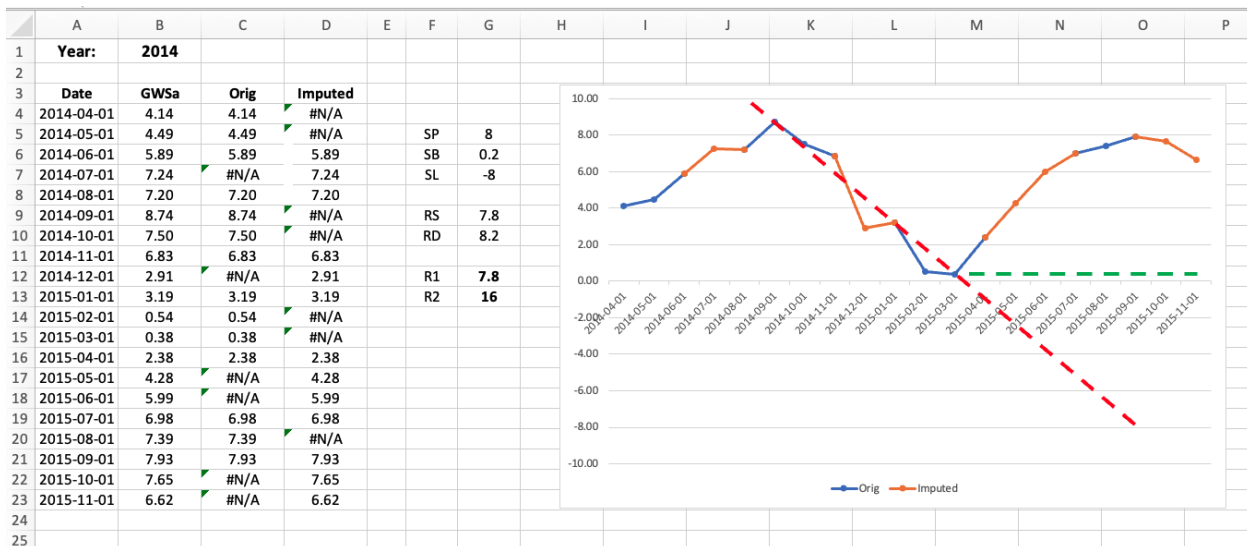


And finally, the gap imputation with 4 trend lines results in the following:



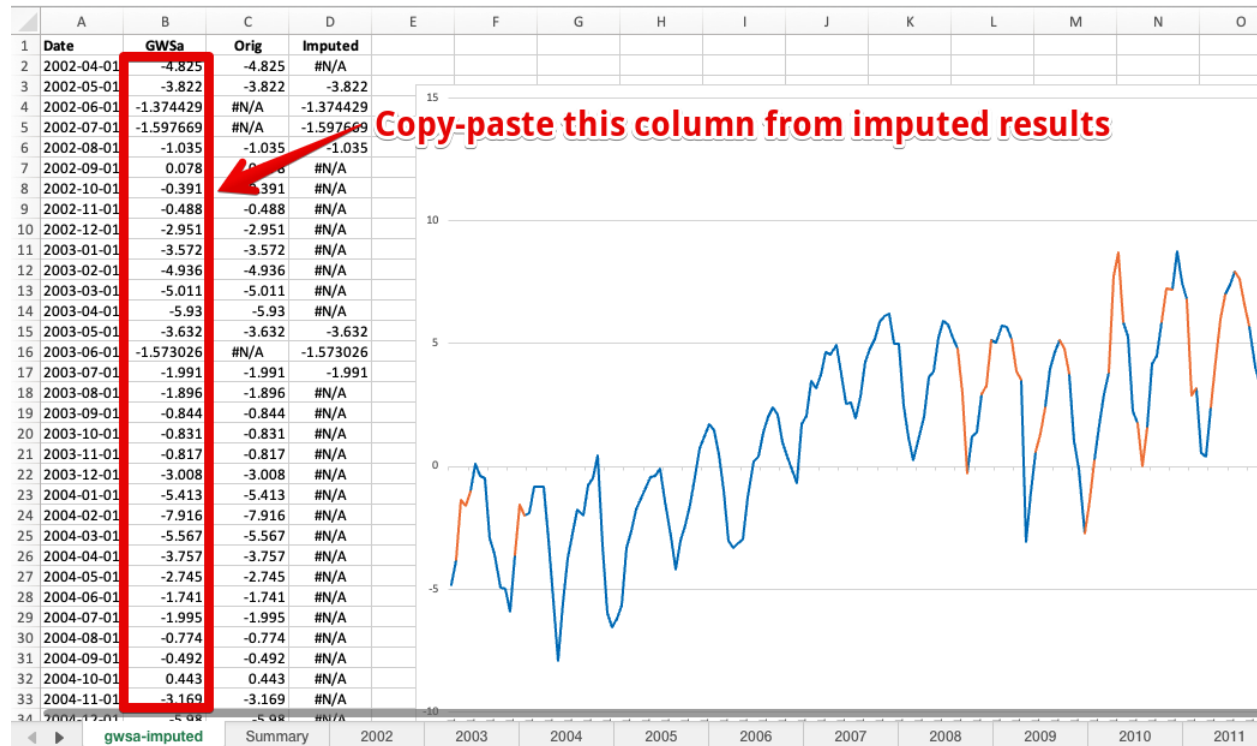
6.7 Data Processing Examples

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 and/or method 2.

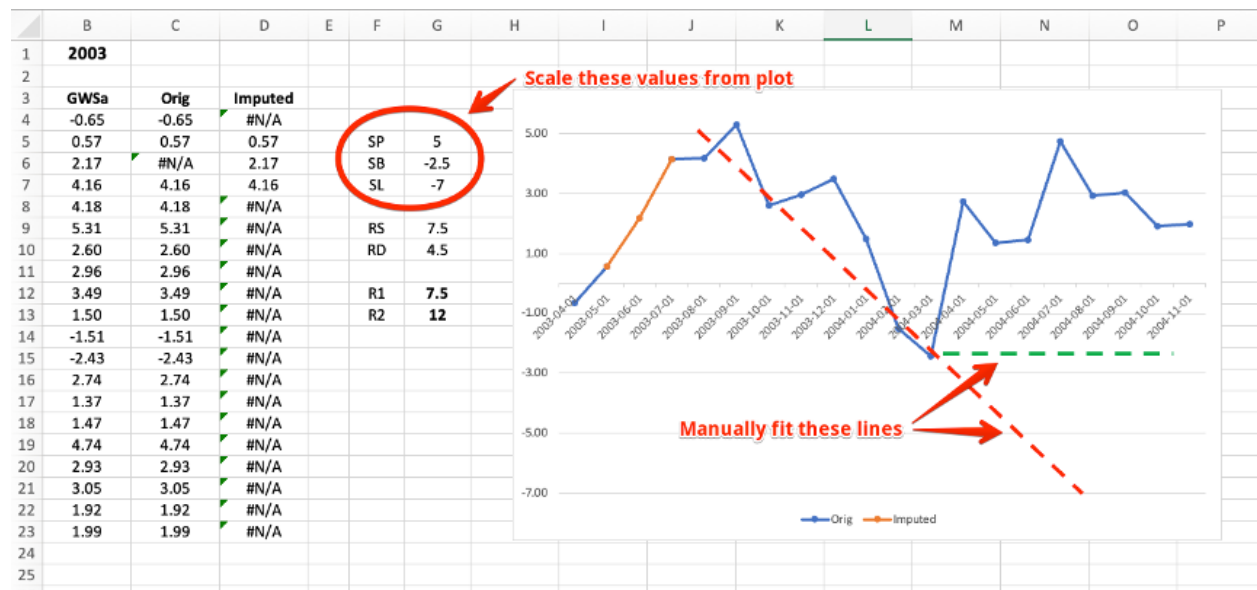


The following Excel file illustrates how to examine and process each season of data from a GRACE-derived and imputed groundwater storage anomaly time series: [west-gwsa-wtf.xlsx](#)

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.

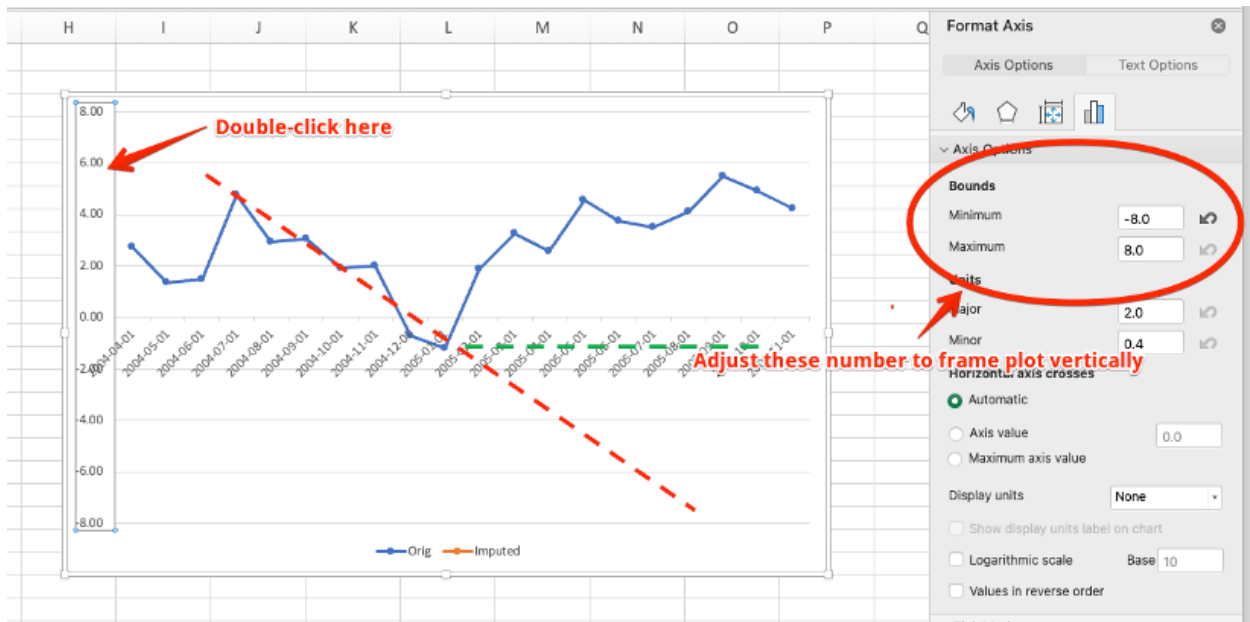


At this point you can browse through each of the tabs for the years starting in 2002. On each page, the seasonal values are automatically pulled from the main sheet using a VLOOKUP formula. For each page, manually adjust the red and green lines to fit the descending branch and the base. Then manually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





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 of the years and calculating all of the R_1 , R_2 values, you can see a summary in the Summary sheet.

