**GNSS2TWS\_Slepian: A software to recover daily GNSS-inverted terrestrial water storage changes based on Slepian basis functions**

***User Manual***

Written by Zhongshan Jiang

Sun Yat-sen University

May 2024

# 1 Installation

The primary purpose of this code is to achieve the recovery of daily large-scale terrestrial water storage (TWS) changes from GNSS-observed crustal vertical positions. Differing from the widely-used spatial-domain inversion strategy based on Green's function method (Jiang et al., 2022b), our inversion modeling is implemented in the spectral domain based on Slepian basis functions, which aims to infer daily large-scale TWS changes using unevenly distributed GNSS vertical data. We assume that the potential users are familiar with the Matlab language programming, PCA decomposition, inversion method, Slepian basis functions, and mass loading theory. The code has been written in MATLAB (tested with version 2018b and 2024a) and has been tested on Windows (version 11) operating systems.

Readers can download the complete software (including the code and one example (all input data)) from GitHub (https://github.com/jzshhh/gnss2tws\_slepian) we also release this open-source tool (including the code and one example (all input/output data)) on Zenodo (https://doi.org/10.5281/zenodo.11206318). There are no additional requirements except for the MATLAB language running environment and some basic toolboxes. In this public edition 1.1 of GNSS2TWS\_Slepian, we provide an example to investigate the spatiotemporal TWS changes in the Western United States. Readers can directly run this example without additional operations. **Note that if your computer doesn’t have enough memory, please assign the variable ‘lmax’ (in file “load\_scenario.m”) with smaller degrees.**

# 2 Software introduction

2.1 Structure of GNSS2TWS

The top folder of ‘gnss2tws\_slepian’ includes 4 subfolders (‘code/’, ‘data/’, ‘result/’ and ‘HATool/’) and 2 Matlab scripts (‘gnss2ewh\_slepian\_main.m’ and ‘load\_scenario.m’) (see Figure 1).

(1) The subfolder ‘code/’ includes all sub-routines for the inversion code, there are 8 subfolders (‘Initialization/’, ‘Data Loading/’, ‘Decomposition/’, ‘Eigen Paraments/’, ‘Inversion/’, ‘Plotting/’, ‘slepian\_alpha/’, and ‘map\_toolbox’. Each subfolder saves the corresponding main driver functions and related sub-routines. Note that the scripts in subfolder ‘map\_toolbox’ are copies of the Matlab built-in functions in toolbox ‘map’. The subfolder ‘slepian\_alpha’ saves the open-source Slepian Alpha software available from GitHub (https://github.com/csdms-contrib/slepian\_alpha).

(2) The subfolder ‘data/’ includes all datasets, including GNSS data associated with hydrological cycles, station information files, and other auxiliary files (e.g., boundary files of the study area).

(3) The subfolder ‘result/’ saves all output results and figures for the final analysis and instruction.

(4) The subfolder ‘HATool/’ provides some practical tools for conventional hydrological analysis (see more details in Text S4).

(5) The ‘load\_scenario.m’ file saves all parameters and most of them need some adjustments to match specific study cases.

(6) The ‘gnss2ewh\_slepian\_main.m’ file is the main program of the ‘gnss2tws\_slepian’.

Figure 1. The directory structure of the inversion code ‘gnss2tws\_slepian’.

2.2 Program flow of GNSS2TWS

The GNSS2TWS\_Slepian software is designed with different structured modules and readers can easily follow the logic of program execution. As demonstrated in Figure 2, the procedure workflow is composed of (1) loading scenario, (2) loading data, (3) PCA decomposition, (4) calculating eigenmatrix, (5) inversion modeling, and (6) displaying results.

1) Loading scenario

Scheme initialization mainly includes two aspects: parameter configuration and directory generation. Parameter configuration includes the assignment of relevant parameters and path settings of necessary files. The initialization also produces some directories that store the intermediate variables and final results.

2) Loading data

All preprocessed GNSS vertical time series stored in each file are collected to build an observation matrix with NaN for missing values. Besides, this step also collects some pertinent information regarding station name and location (i.e., longitude and latitude).

3) PCA decomposition

The built-in ALS-based PCA function is called to decompose the previously generated observation matrix and two groups of matrixes (i.e., spatial and temporal functions) are produced in this step. The determination of the PC’s number generally depends on that gradually increasing the PC’s number does not noticeably improve the fit to the raw data.

4) Calculating eigenmatrix

This step calls SLEPIAN Alpha (Harig et al., 2015) to calculate the eigenvalues and eigenvectors as described in Text S1, which only rely on the specific study areas. The eigenvectors are used for the production of Slepian basis functions.

5) Inversion modeling

In our inversion, the spatial functions of each PC are converted into spectral-domain VCD-related Slepian coefficients and are then transformed into EWH-related Slepian coefficients using load Love numbers (Wang et al., 2012). After that, grided EWH values for each PC are calculated and then multiplied by the corresponding temporal function, and the total EWH changes at all grids are synthesized by summing the foregoing products of all PCs.

6) Displaying results

To quickly view inversion results, some figures are plotted for readers. This step displays figures of spatial and temporal functions, maps of annual EWH amplitudes, maps of selected Slepian basis functions, and figures of eigenvalues as the degree changes.



Figure 2. Program workflow and main modules of GNSS2TWS\_Slepian.

2.3 Hydrological analysis tools

In the folder of ‘HATool’, the GNSS2TWS\_Slepian software provides some practical tools for conventional hydrological analysis. The main functions include the calculation of basin-averaged EWH time series, climatological averages, EWH anomaly time series, water storage deficit, and surplus time series, and standardized EWH anomaly time series. The main functions ‘plot\_gnss\_area\_ewh\_series\_main.m’ and ‘plot\_gnss\_drought\_characterization\_main.m’ should be adjusted to make available for their interested areas.

**1) Calculating basin-averaged EWH time series**

The basin-averaged EWH time series are calculated based on a latitude-weighted scheme and it contributes to analyzing time-varying characteristics of watershed-scale TWS changes. The script ‘plot\_gnss\_area\_ewh\_series\_main.m’ calls the function ‘cal\_basin\_average\_time\_series’ to calculate basin-averaged EWH time series within known boundaries and displays GNSS-based time-varying water estimates (see examples in Figure 3).

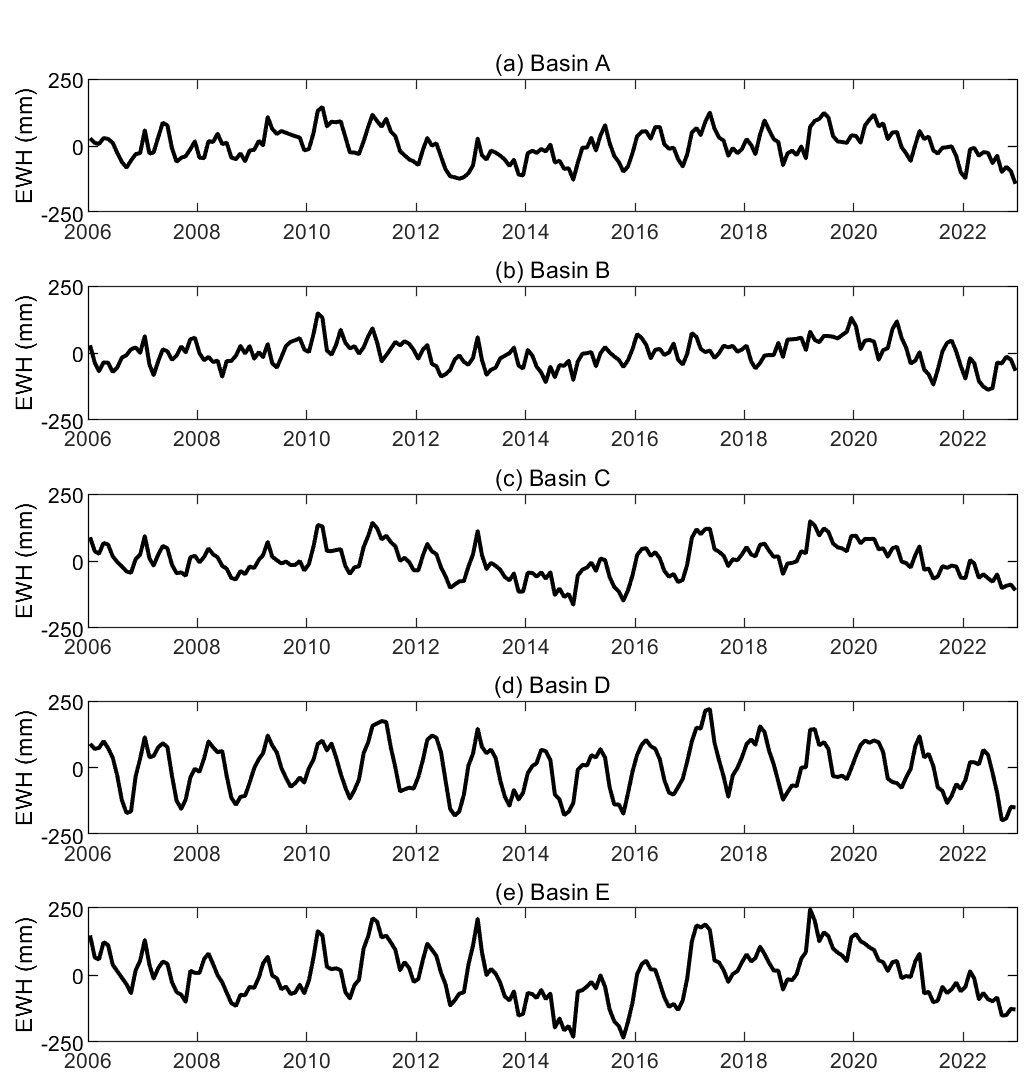


Figure 3. GNSS-based time-varying water estimates in five river basins.

**2) Characterization of hydrological extremes**

Methods in climatology are used for the characterization of hydrological extremes, i.e., the GNSS inversion results are used for drought assessment using the GNSS-based drought severity index (GNSS-DSI) (Jiang et al., 2022a). The script ‘plot\_gnss\_drought\_characterization\_main.m’ calls ‘cal\_gnss\_dsi.m’ to calculate climatological averages, EWH anomaly time series, water storage deficit, and surplus time series, and standardized EWH anomaly time series (see examples in Figure 4).

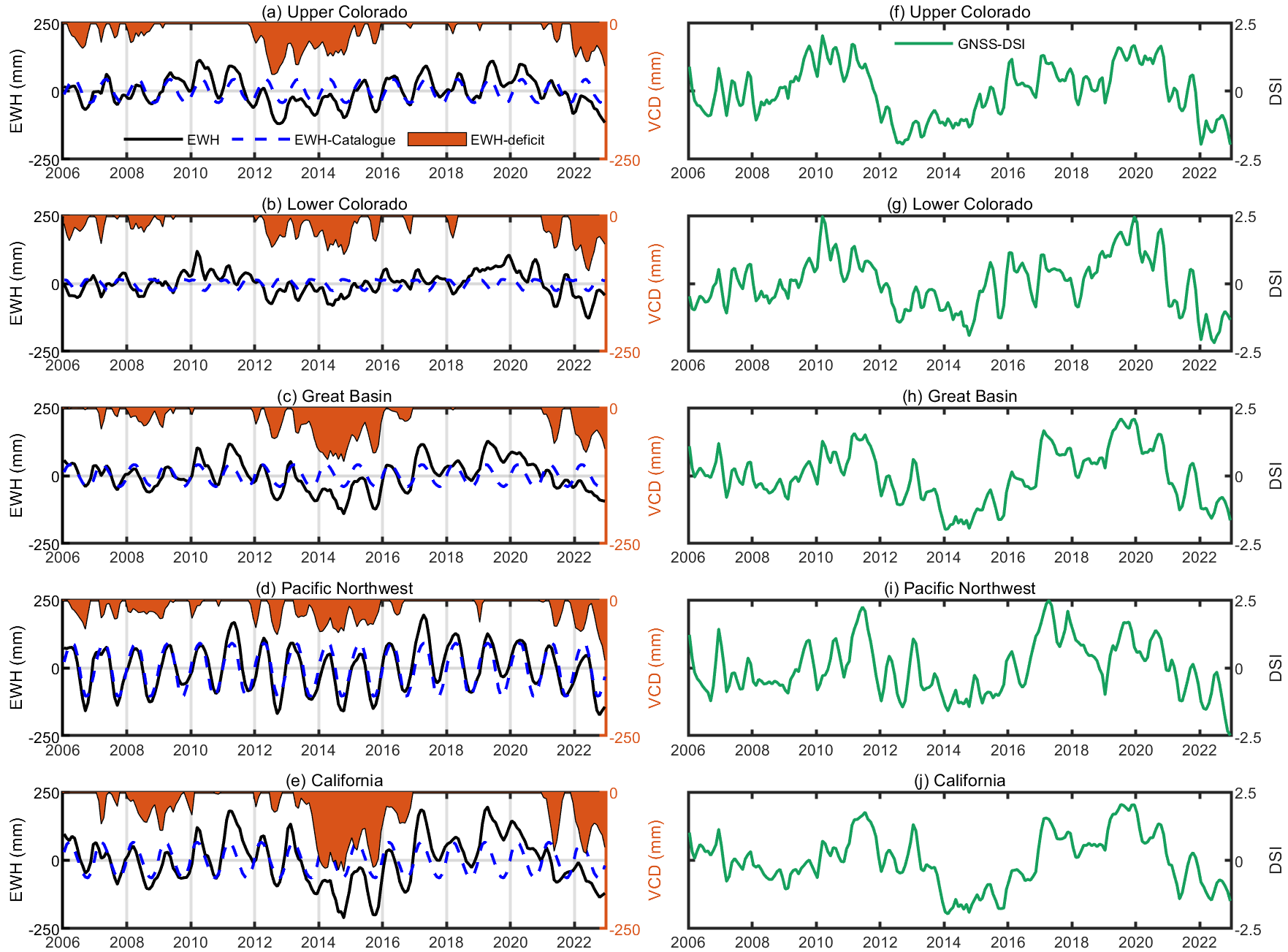


Figure 4. GNSS characterization of hydrological droughts in five drainage basins. (a–e) indicate GNSS-derived water storage deficits. Black solid lines indicate the monthly EWH anomalies, Blue dashed lines represent the climatological normal, and red shadings denote the EWH deficits. (f–j) GNSS-DSI (green line) results.

# 3 Example

To demonstrate the performance of GNSS2TWS\_Slepian, we investigate the large-scale TWS changes and characterize hydrological extremes based on the inversion of daily GNSS vertical crustal displacement in the Western United States.

3.1 Preparing data

**1) GNSS data**

* **Download GNSS data**

The GNSS position time series in the western United States are downloaded from the Nevada Geodetic Laboratory, University of Nevada, Reno, United States (http://geodesy.unr.edu/gps\_timeseries/tenv3\_loadpredictions). The new enhanced version of GNSS products includes the predictions of non-tidal atmospheric loading, and non-tidal ocean loading products produced by the German Research Centre for Geosciences. Readers can download their interested data at each station from http://geodesy.unr.edu/gps\_timeseries/tenv3\_loadpredictions/????.tenv3 (note that ???? is a 4-char name for each GNSS station). Brief documentation for the ‘tenv3’ format of the NGL GNSS time series product is summarized at http://geodesy.unr.edu/gps\_timeseries/README\_tenv3load.txt. The daily solutions at 332 stations are chosen according to data continuity, uncertainty, length, and deformation characteristic (e.g., poroelastic and elastic response). More details about the criteria for station selection can be found in previous studies (Argus et al., 2021). The well-modeled predictions including non-tidal oceanic loading and atmospheric loading effects are subtracted from the previously downloaded GNSS time series. The used data records are then saved in the ‘\*.pos’ file and the format of each row in the ‘\*.pos file is “date (yyyymmdd) N (mm) E (mm) U (mm) sig\_n (mm) sig\_e (mm) sig\_u (mm)” (Figure 3)”. These ‘\*.pos’ files are the input files of the ‘lsf’ code (https://github.com/jzshhh/lsf).

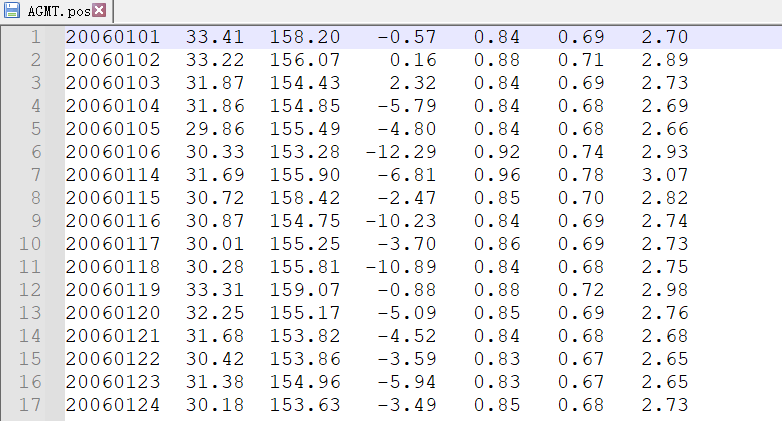


Figure 5. Example of data records in each ‘\*.pos’ file.

* **Extracting hydrological loading displacement**

After removing the non-tidal oceanic loading and atmospheric loading effects, the GNSS vertical time series are then modeled with a constant, a linear (indicating long-term tectonic motion), two sinusoids (representing annual and semiannual signals), and some Heaviside (representing instrumental and coseismic jumps) and logarithmic (modeling postseismic deformation) functions when needed. We estimate all parameters based on a least-squares fitting method and remove the long-term linear trend, offsets, and postseismic deformation to obtain vertical position time series dominated by seasonal hydrological loading deformation. The Matlab code to extract surface displacement associated with water cycles is also released on GitHub (named ‘lsf’, https://github.com/jzshhh/lsf), which is a modified version of Tsview software (<http://www-gpsg.mit.edu/~tah/GGMatlab/#_tsview>) and is suitable for batch processing of GNSS time series.

* **Saving GNSS data recognized by the software**

The residual vertical time series at each station is saved in a separate ‘\*.up” file with a 4-char station name. The format of each row in the ‘\*.up’ file is “date (yyyymmdd) up (mm) sig\_up (mm)” (Figure 6). All files are saved in the folder ‘data/gps/’.

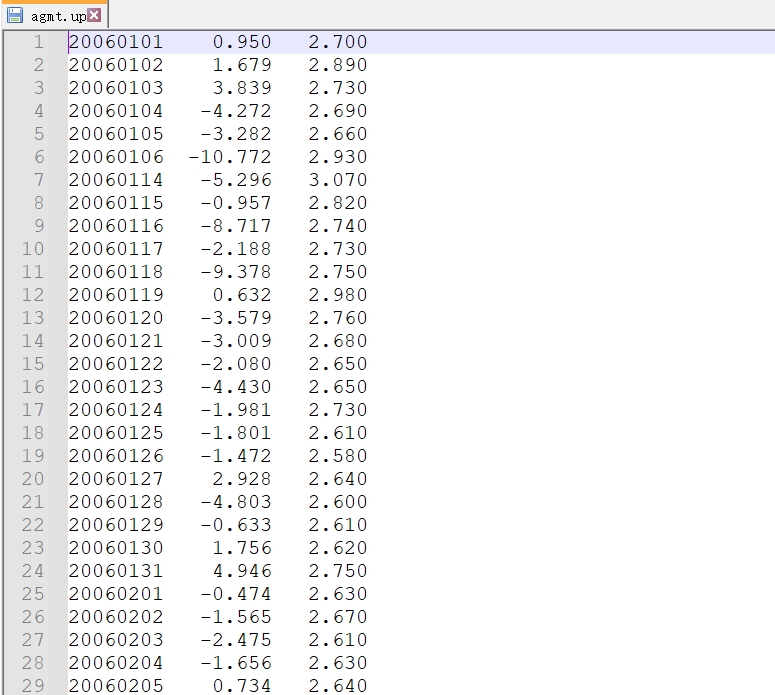


Figure 6. Example of data records in each ‘\*.up’ file.

**2) Boundary file**

In the subfolder ‘data/’, we provide two functions ‘making\_boundary.m’ and ‘making\_extended\_boundary.m’ for readers to generate boundary files (e.g., ‘USA\_border\_L1.txt’) and extended boundary dataset (e.g., ‘USA\_border\_L2.txt’) to reduce inversion artifacts near the edges (Figure 7). The contents of these two boundary files are shown in Figure 8.

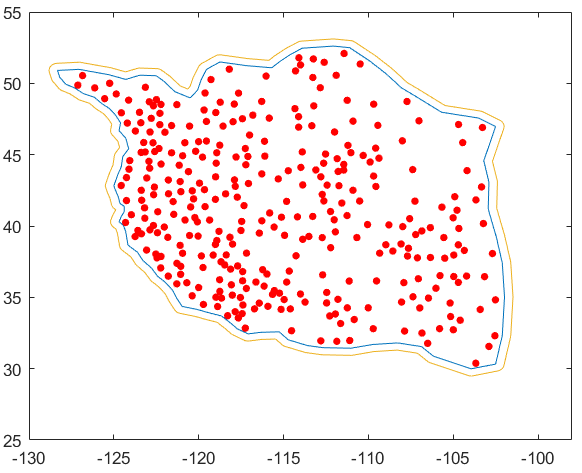


Figure 7. Boundary files of the Western United States. The blue line is our selected boundary. The outer orange lines are boundary extended with a buffer. Red solid circles are the locations of GNSS stations.

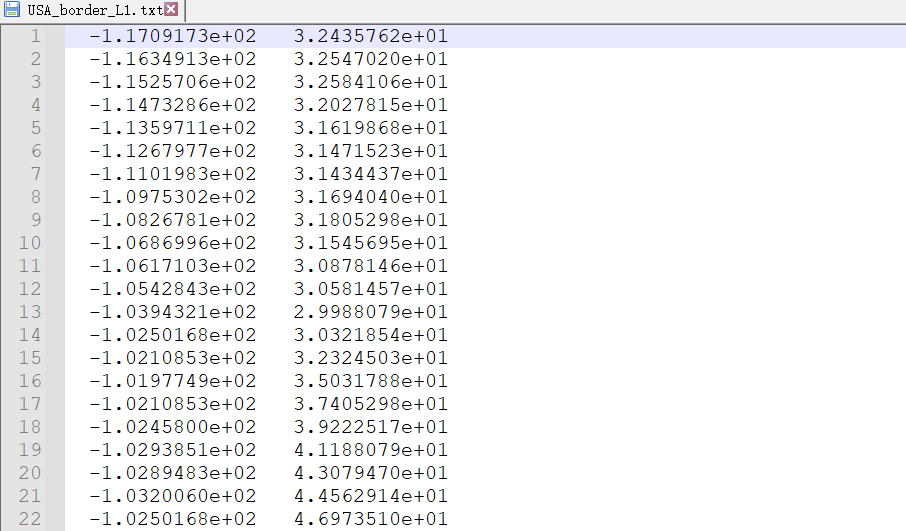


Figure 8. Data records in the boundary file (‘USA\_border\_L1.txt’’).

**3) Station information file**

The station information is saved in the file ‘sites.all’, the format in each row is ‘name, longitude, and latitude’ (Figure 9). Note that all GNSS data saved in the directory “data/gps/” need have station information.

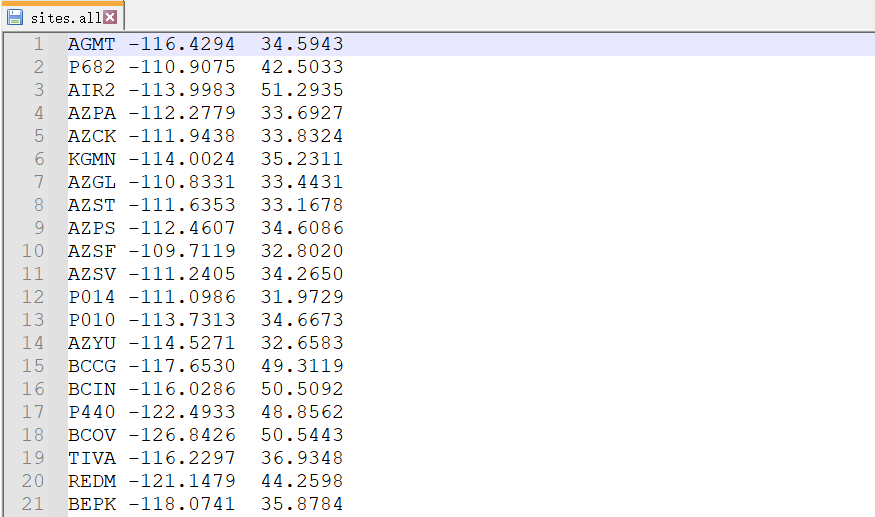


Figure 9. Data records in the station information file (‘sites.all’).

3.2 Running the Program

After preparing all files, we should set up some parameters for our specified model, click Matlab’s Run button, and wait for the result. Here, we will show how to configure the file “load\_scenario.m” and demonstrate some key steps of our inversion model.

**1) Parameter setting**

First, we need to configure the parameters in the file “load\_scenario.m”, which includes information about ‘Constant variable’, ‘GNSS data’, ‘Study area’, ‘PCA Decomposition’, ‘Inversion’, and ‘Plotting’. The example of the file “load\_scenario.m” is shown in Figure 10.

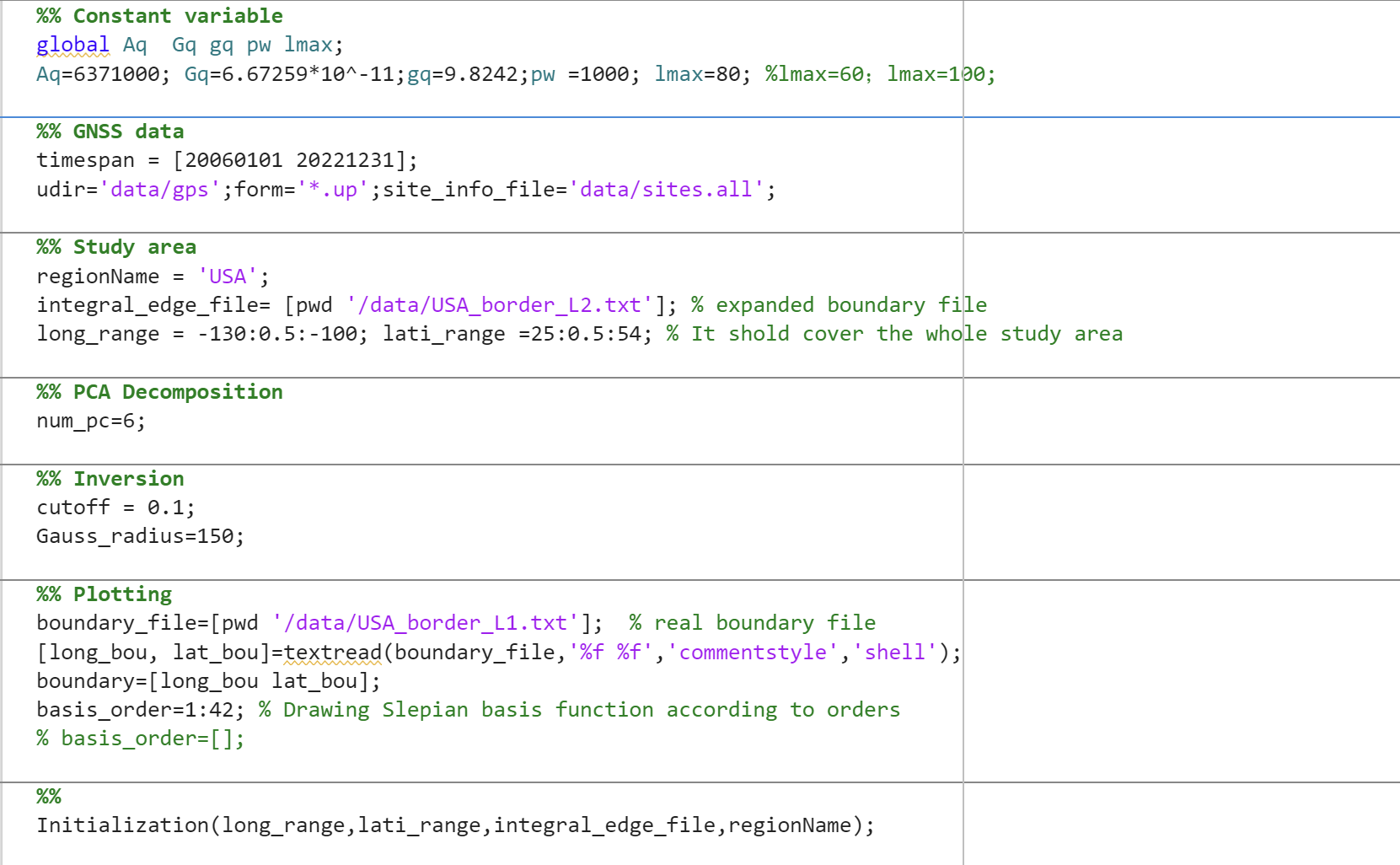


Figure 10. Parameters that need to configure in the file “load\_scenario.m”.

* **Constants**

There are several global variables with constants for the Earth’s average radius (Aq=6371000 m), universal gravitational constant (Gq=6.67259\*10^-11 N·m²/kg²); gravitational acceleration (gq=9.8242 m/s2), water density (pw =1000 kg/m3), and maximum degree of spherical harmonics (lmax = 80). **Note that if your computer doesn’t have enough memory, please assign ‘lmax’ with smaller degrees.**

* **GNSS data**

The study period should be set for reading GNSS data. In this example, our study period is from 2006-01-01 to 2022-12-31, so we set “timespan=[20060101 20221231];”.

The path of GNSS data, file extension, and GNSS site information file should be correctly located, e.g., “udir='data/gps; form='\*.up'; site\_info\_file='data/sites.all;”.

* **Study area**

This step aims to set up parameters for region name, integral boundary file, and range of study area. There are several parameters:

‘regionName’ represents the study region name, e.g., regionName = 'USA';

‘integral\_edge\_file’ indicates the path of the extended boundary data, e.g., integral\_edge\_file= [pwd '/data/USA\_border\_L2.txt'];

‘long’ indicates longitude range, e.g., long\_range = -130:0.5:-100;

‘lat’ indicates latitude range, e.g., lati\_range =25:0.5:54;

* **PCA decomposition**

This aims to set up the number of selected PCs for inferring principal EWH components. The number is generally determined when the improvement of fit to the GNSS data is insignificant with more PCs. e.g., “num\_pc=6;”.

* **Inversion**

Two parameters need to be set up for inversion.

‘cutoff’ shows the threshold value for an energy concentration degree, e.g., cutoff = 0.1;

‘Gauss\_radius’ shows the Gaussian filtering radius, e.g., Gauss\_radius=150;

* **Plotting**

The below parameters are configured for plotting:

boundary\_file=[pwd '/data/USA\_border\_L1.txt']; % selected boundary file

[long\_bou, lat\_bou]=textread(boundary\_file,'%f %f','commentstyle','shell');

boundary=[long\_bou lat\_bou];

% basis\_order=1:42; % Drawing Slepian basis function according to orders

basis\_order=[];

**2) Key steps of our inversion model**

After preparing all files and setting up all parameters, we only need open the main program “gnss2ewh\_slepian\_main.m”, click Matlab’s Run button (Figure 11), and wait for the result. The program can automatically run without any user intervention and some key steps are designed with dialog/interface boxes to show which step the program is executing (Figures 12–16).

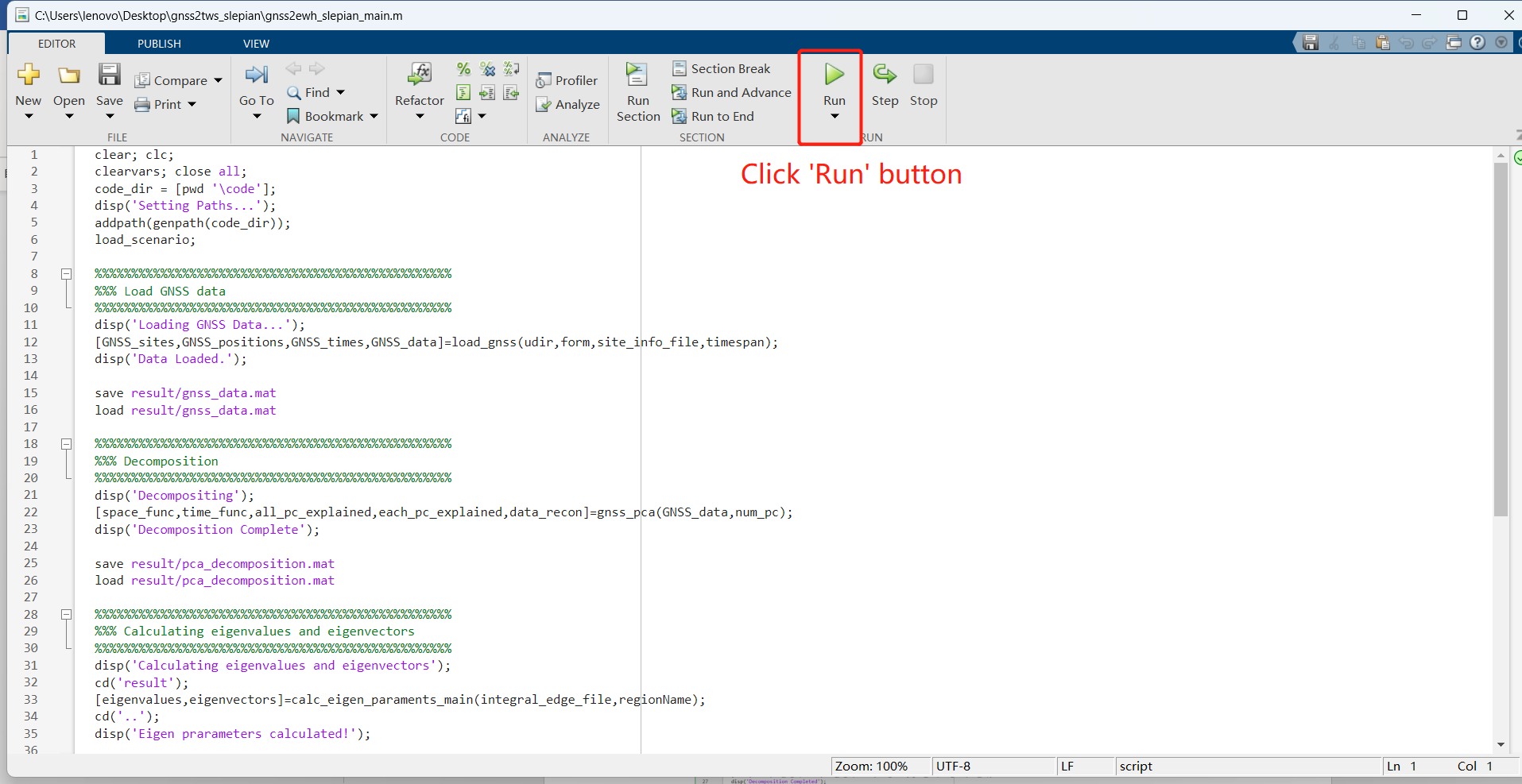


Figure 11. Executing the main program.

* **Scheme initialization**

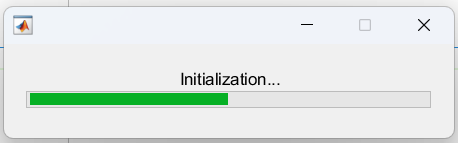


Figure 12. Scheme initialization. It realizes the parameter configuration and directory generation.

* **Loading GNSS data**

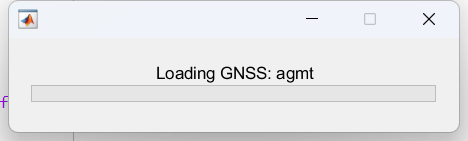


Figure 13. Loading GNSS data.

* **PCA decomposition**

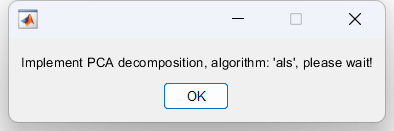


Figure 13. PCA decomposition of GNSS data.

* **Calculating eigenmatrix**

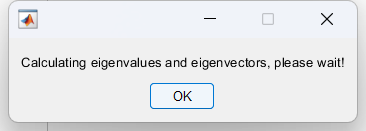


Figure 14. Calculating the eigenvalues and eigenvectors.

* **Calculating Laplacian matrix**

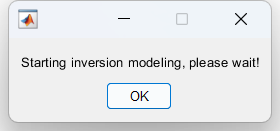


Figure 15. Performing the inversion process.

* Displaying results

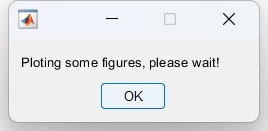


Figure 16. Generating some figures to quickly view inversion results.

3.3 Displaying inversion results

After performing the inversion, some results (in the form of ‘.mat’ and ‘\*.tiff’) are saved in the folder “gnss2tws\_slepian\result”. The destination is to display the results (Figures 17–21) and readers need to replot these figures for scientific publication using corresponding data sets.

* **Temporal and spatial functions**

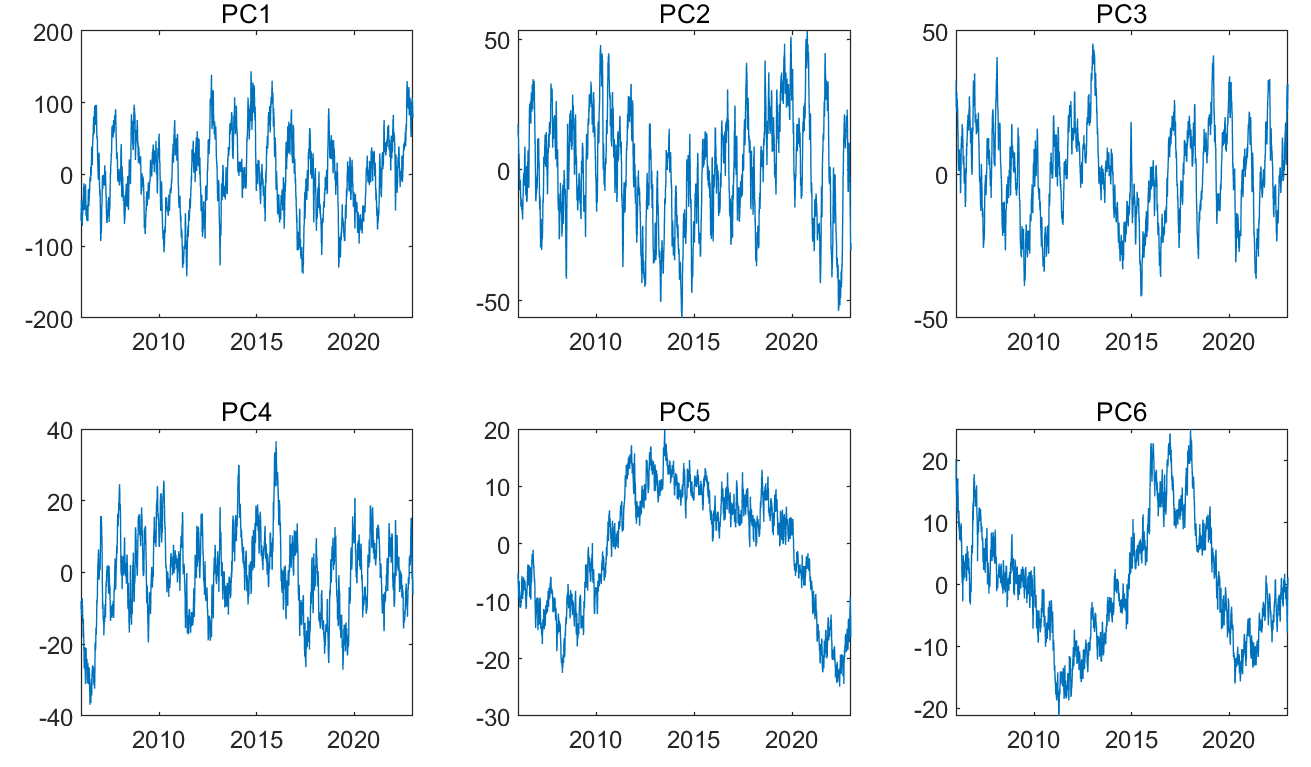


Figure 17. Temporal functions of the first six PCs.

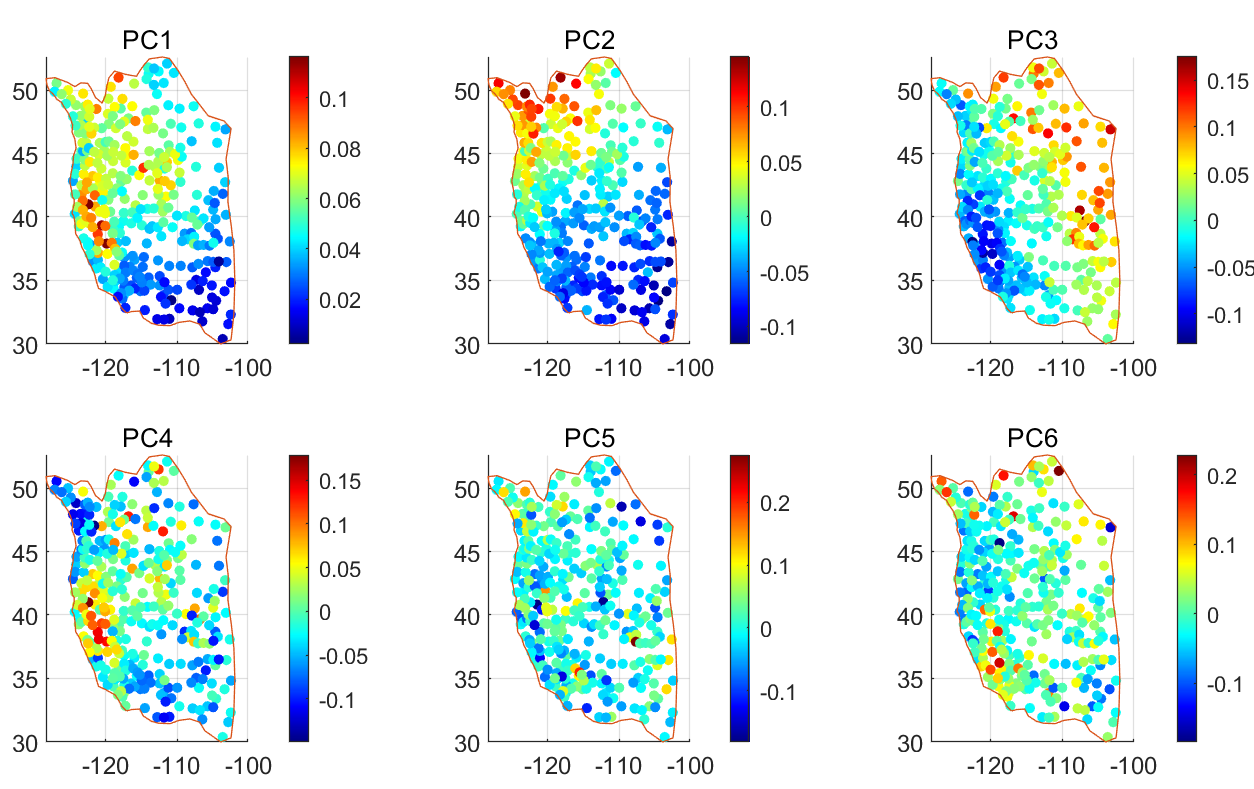


Figure 18. Spatial functions of the first six PCs.

* **Results of Slepian basis functions**

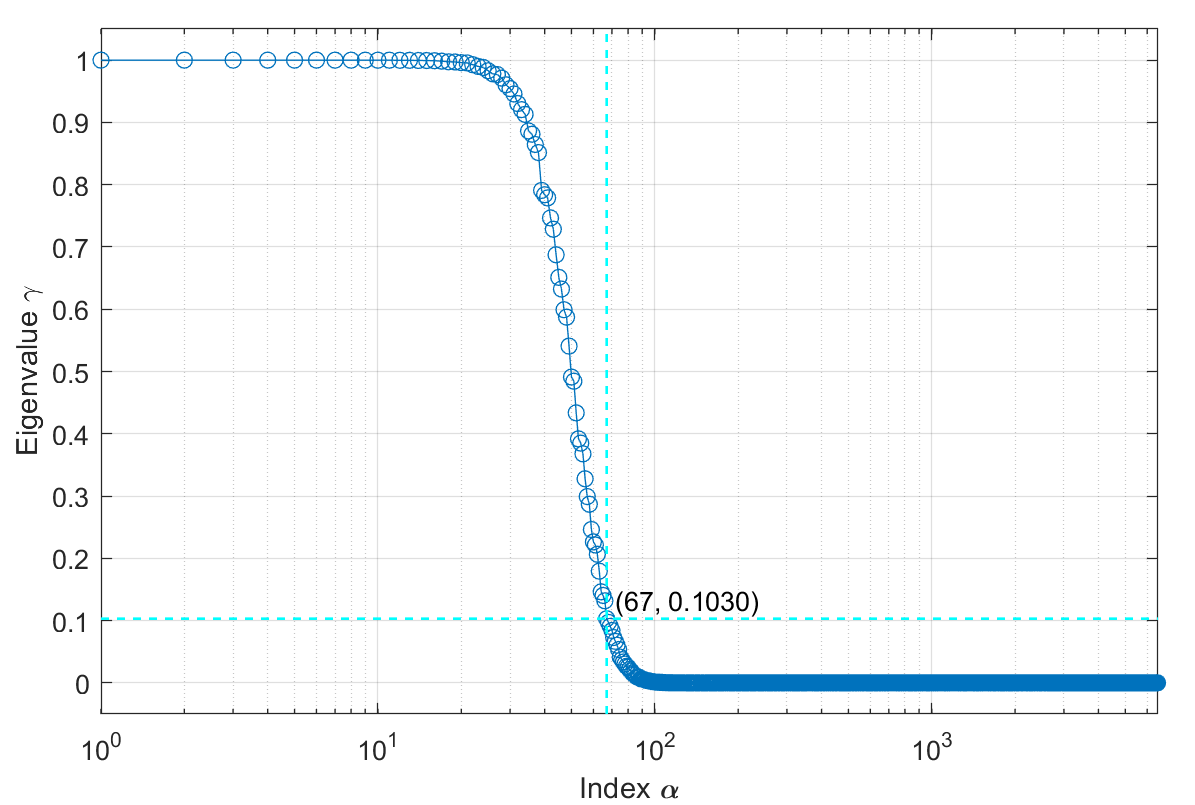


Figure 19. The concentration ratio of all Slepian basis functions corresponding to spherical harmonic functions with a given bandwidth.

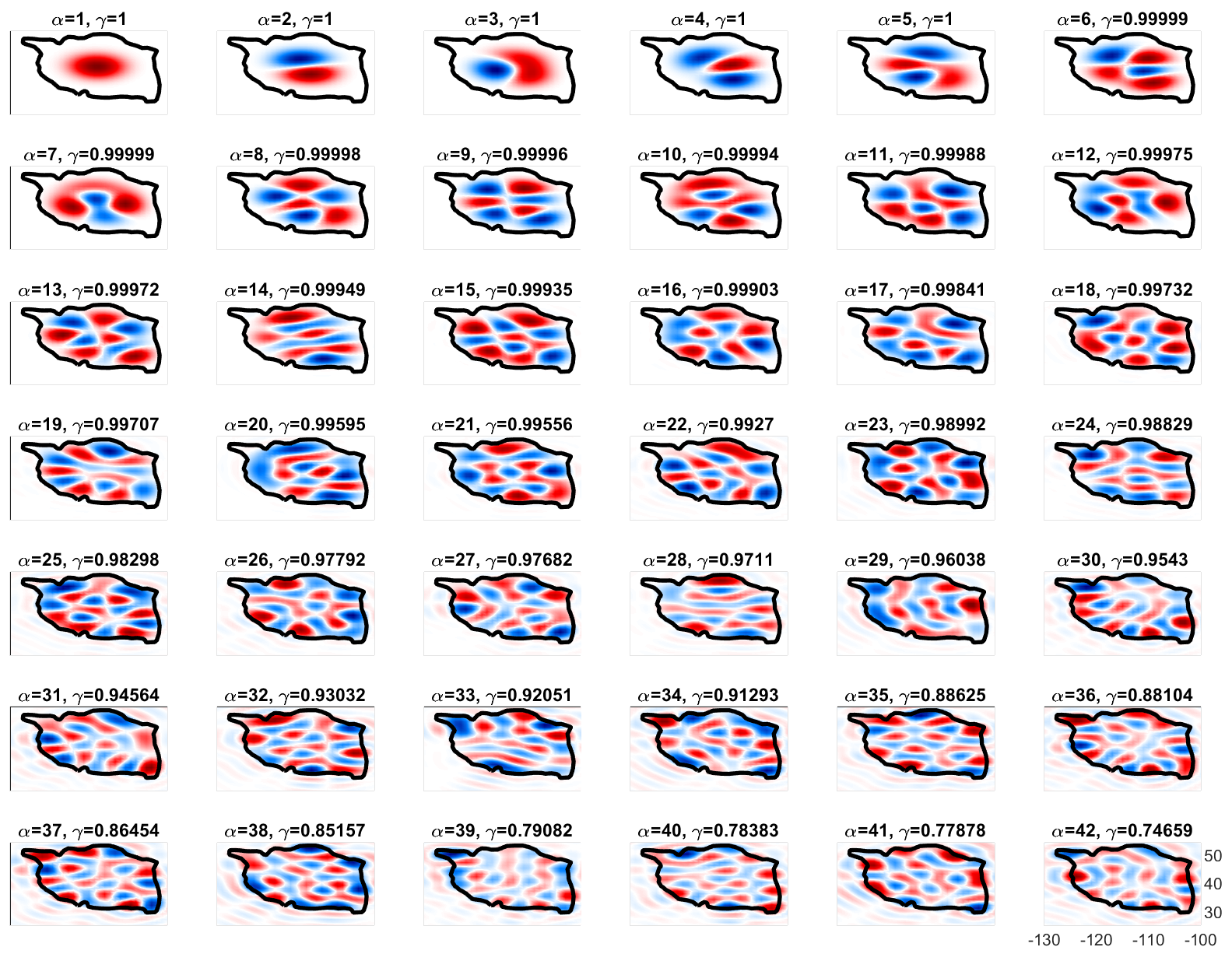


Figure 20. Examples of the first 42 Slepian basis functions with black boundary lines.

* **Annual amplitudes of GNSS-based EWH**

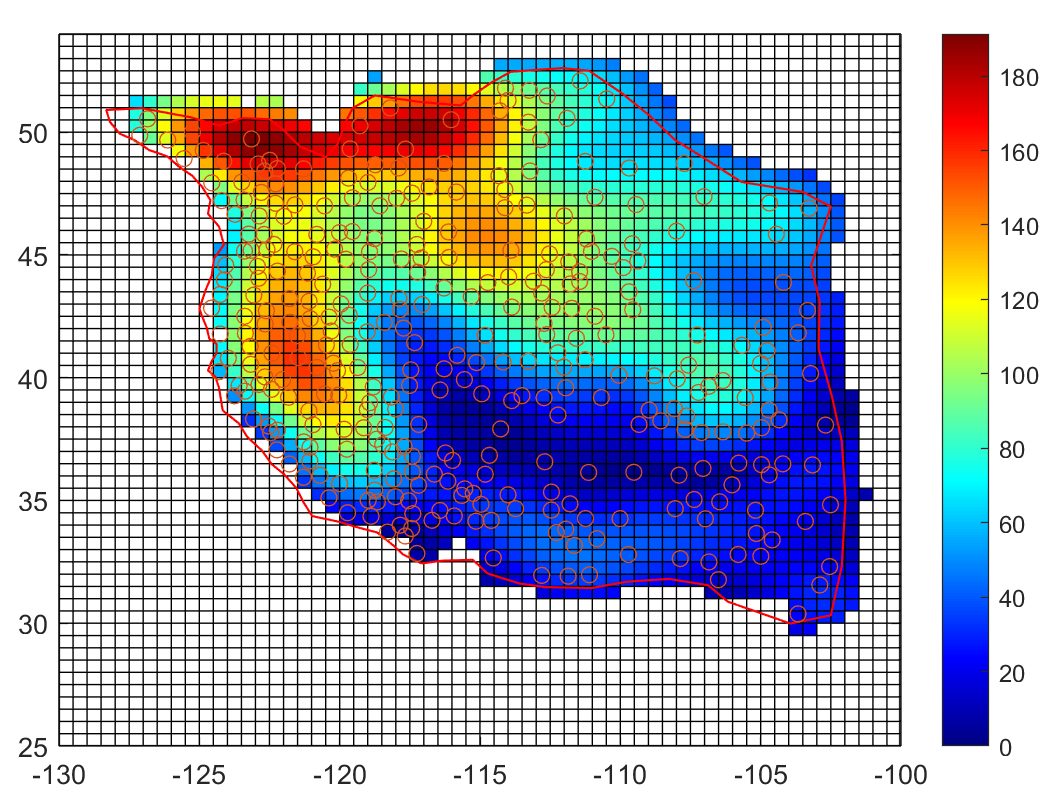


Figure 21. Map of annual amplitudes of GNSS-inverted EWH.

# 4 Contact

This is the public version 1.1 of GNSS2TWS\_Slepian. Readers can modify the code to make it available for their applications. Please do not hesitate to contact us if you come across any bugs or have any comments, suggestions, or corrections. The contact details are as follows:

Name: Zhongshan Jiang

E-mail: jiangzhsh@mail.sysu.edu.cn

Address: School of Geospatial Engineering and Science, Sun Yat-sen University, Zhuhai 519082, China

References

Argus, D.F., Peltier, W.R., Blewitt, G., Kreemer, C., 2021. The Viscosity of the Top Third of the Lower Mantle Estimated Using GPS, GRACE, and Relative Sea Level Measurements of Glacial Isostatic Adjustment. Journal of Geophysical Research: Solid Earth 126.

Harig, C., Lewis, K.W., Plattner, A., Simons, F.J., 2015. A Suite of Software Analyzes Data on the Sphere. Eos 96.

Jiang, Z., Hsu, Y.-J., Yuan, L., Cheng, S., Feng, W., Tang, M., Yang, X., 2022a. Insights into hydrological drought characteristics using GNSS-inferred large-scale terrestrial water storage deficits. Earth and Planetary Science Letters 578.

Jiang, Z., Hsu, Y.-J., Yuan, L., Feng, W., Yang, X., Tang, M., 2022b. GNSS2TWS: an open-source MATLAB-based tool for inferring daily terrestrial water storage changes using GNSS vertical data. GPS Solutions 26.

Wang, H., Xiang, L., Jia, L., Jiang, L., Wang, Z., Hu, B., Gao, P., 2012. Load Love numbers and Green's functions for elastic Earth models PREM, iasp91, ak135, and modified models with refined crustal structure from Crust 2.0. Computers & Geosciences 49, 190-199.