USER MANUAL

Angel Martin[1], Raquel Luján, Ana Belén Anquela

Department of Cartographic Engineering, Geodesy and Photogrammetry
Univesitat Politècnica de València

[1]Contact: aemartin@upvnet.upv.es

A package of python software tools for GNSS interferometric reflectometry (GNSS-IR) is provided. Although we refer to GNSS, these first version of the software package only works with GPS satellite constellation. The software works in Python 2.7 and Python 3.

Installation

The software is written in python, so python programming language is needed to be installed in the user's computer. Also, datetime, os, numpy, math, matplotlib and astropy libraries are needed, so the users should install them.

There is no need to install anything more, only download and decompress the *SNRpy* file. The decompression of the *SNRpy* file should produce the scheme of folders and files which can be seen in Figure 1.

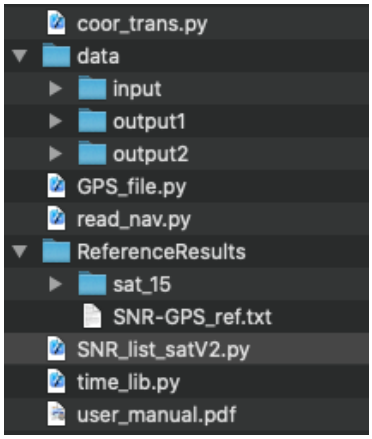


Figure 1. Folders and files from the decompression of *SNRpy* file

There are two folders (*data* and *ReferenceResults*), five *.py* software files and the *user_manual.pdf* file containing this user manual.

Data folder contains 3 folders, *input* folder with 7 observation and 7 navigation RINEX files, Figure 2, so that the user can validate the software operation, and two empty folders, *output1* and *output2*.

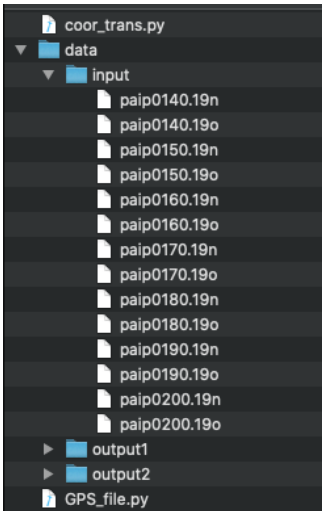


Figure 2. Files in the folder input.

ReferenceResults folder, Figure 3, is a folder where files are stored to check that the results obtained by the user are correct. This folder contains the file *SNR-GPS_ref.txt* with the results of the first software module and the folder *sat_15* which contains the plot files and output text files for the results of the second software module regarding GPS satellite number 15.

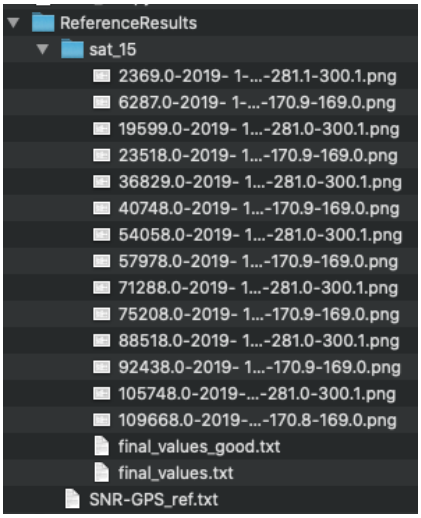


Figure 3. Files and folders in the ReferenceResults folder.

Checking the software

To check the software, GNSS data are provided. This data belongs to the experiment performed in the installations of the Cajamar Center of Experiences, Paiporta, Valencia, Spain (Martin el al. 2019). We provide 7 days (from a total of 66 in the experiment) of

GNSS observations with a geodetic GNSS receiver (Trimble R10, from Department of Cartographic Engineering, Geodesy and Photogrammetry of the Universitat Politècnica de València). RINEX observation (with a 5 seconds sample rate) and navigation files are provided. L1 frequency is used for the example.

The observations files included with the software are daily data files from January 14 to January 20, 2019 and the height of the antenna to the ground was 1.8 m.

There was no occurrence of rain during the observation.

First step

The first step consists in run *GPS-file.py* software file.

This software transforms the RINEX version 3 format observation and navigation files to a unique output file containing the epoch of each observation, the satellite identification, SRN observation, and computed azimuth and elevation of the satellite from the navigation RINEX file. A line in the output file is written only if the satellite elevation falls between 5 and 30 degrees. The input can be a single file (one observation day, for example) or several (one week or moth of continuous observation data separated by daily files). The output is always a single file, called SNR-GPS.txt as a default name and located in the */output1/* folder.

The user has to decide the frequency to be used prior to running the software; this can be done by opening a RINEX observation file and inspecting the header information. Once the frequency has been selected, the user should determine the observations related with this frequency and stored in the observation RINEX file that will be used in the process. Those observations are the SNR and the pseudo-range or code observation related to the selected frequency, Figure 4.

```
23
24 ####USER INPUT PARAMETERS
25 path='data/'#directory with the input folder and the output folder
26 ##in the input folder the RINEX observations files are stored and the output folder
27 ##contains the output files
28 pseu='C1C'#This is the code/pseudorange observation used for satellite orbit determination
29 se='S1C'#The user should decide the frequency for SRN observation that will be stored in the output file
30 ####END USER INPUT PARAMETERS
31
32 system='G'#G for GPS satellite constellation
33 if system=='G':
34     nav_sys='n'
35     fichero_sal=open(path+'output1/SNR-GPS.txt','w')
36     fichero_sal.write('epoch;epoch_time;sat;SNR;Elev;Azi;year;month;day \n')
37
```

Figure 4. Input parameters for *GPS-file.py* software, C1C and S1C are the input observations used in this experiment.

The algorithm reads the observation and navigation files stored in a folder *(\data\input* by default). The names and extensions of those files should follow the RINEX standard format: the name is composed of four characters for the station name, three for the day of the year and one character to identify the session. The extension is composed of two characters corresponding with the last two digits of the year and an “o” for the

observation files or “n” for the GPS navigation file. The algorithm orders the input files in chronological sequence and opens them in order.

This software makes a call to *read_nav.py* module which computes the azimuth and elevation of the GPS satellite from the GPS antenna. *Read_nav.py* module calls modules *time_lib.py* and *coor_trans.py*. Therefore, the user only needs to run *GPS-file.py* module.

The algorithm reads each GPS satellite observation for every epoch for every observation file, and writes in the output file (located in the *data**output1*\ folder by default):

1. A numerical identifier related with the epochs of the observed file. This integer identifier number starts with zero for all observations of the first epoch of the first observation file, increases one by one, and ends with the last epoch of the last observation file. This numerical number can be used as the identifier in the second software module. The software writes as many lines, with the same numerical identifier, as there are observations from different satellites in the same epoch.
2. The time related with the previous lines. The year, month, day and a float number containing the hour (calculated with the integer hour, integer minute and float seconds information from the RINEX observation file).
3. The two-digit satellite GPS numerical identifier.
4. The observed SNR.
5. Based in the observation epoch, the pseudo-distance or code observation to the satellite, the station coordinates located in the header of the observation file and the navigation file, the algorithm computes de azimuth and elevation of the satellite from the antenna. This calculation must determine the emission time and the satellite coordinates in the Earth-Fixed-Earth-Centered (ECEF) reference system. For the emission time an iterative “pseudo-range-based algorithm” is used (Sanz et al. 2013). The algorithm described in Leick et al. (2015) on pg. 240 is used for the ECEF satellite coordinates computation. Finally, the Earth rotation during the signal travel is taken into account to obtain the final satellite coordinates in the receiver time. Based on the ECEF coordinates of the antenna and the satellite, the azimuth and elevation are computed and stored in the output file. The module that computes the azimuth and elevation (*read_nav.py*) requires two extra modules, one to compute the Julian and GPS time (*time_lib.py*) and other to compute the spherical geodetic coordinates form the geocentric coordinates and to compute the azimuth and elevation from the geodetic coordinates of the antenna and the satellite (*coor_trans.py*).

To check the results, the output file (*SNR-GPS.txt* file stored in */data/output1/* folder by default) should be the same as can be found in the *ReferenceResuls* folder (*SNR-GPS_ref.txt* file). Figure 5 is a plot of the first lines of the *SNR-GPS_ref.txt* file.

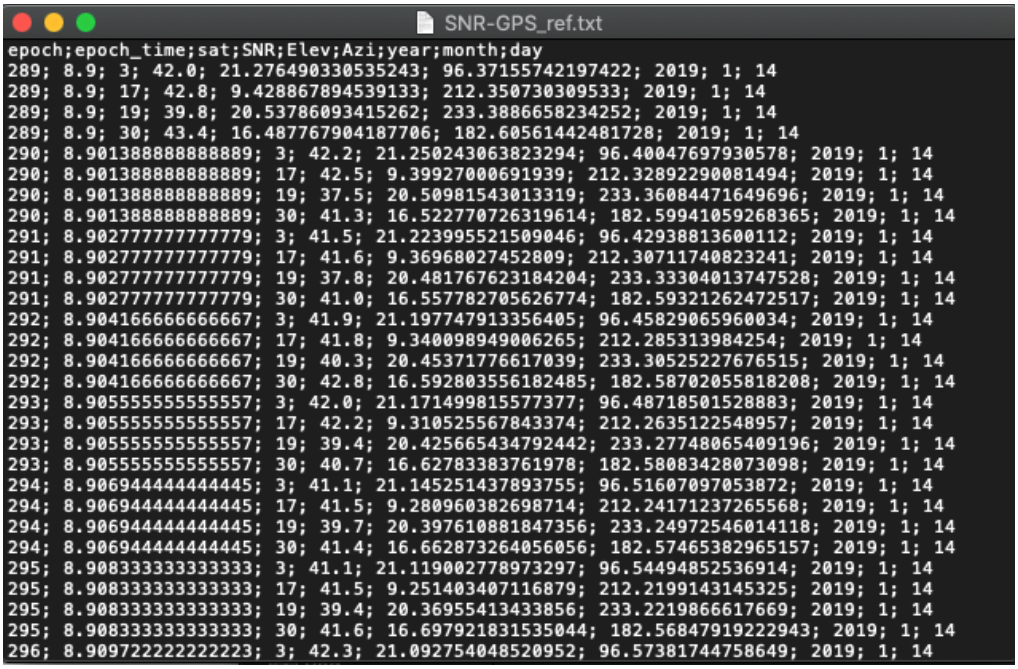


Figure 5. First lines of the output file of the first step.

Second step

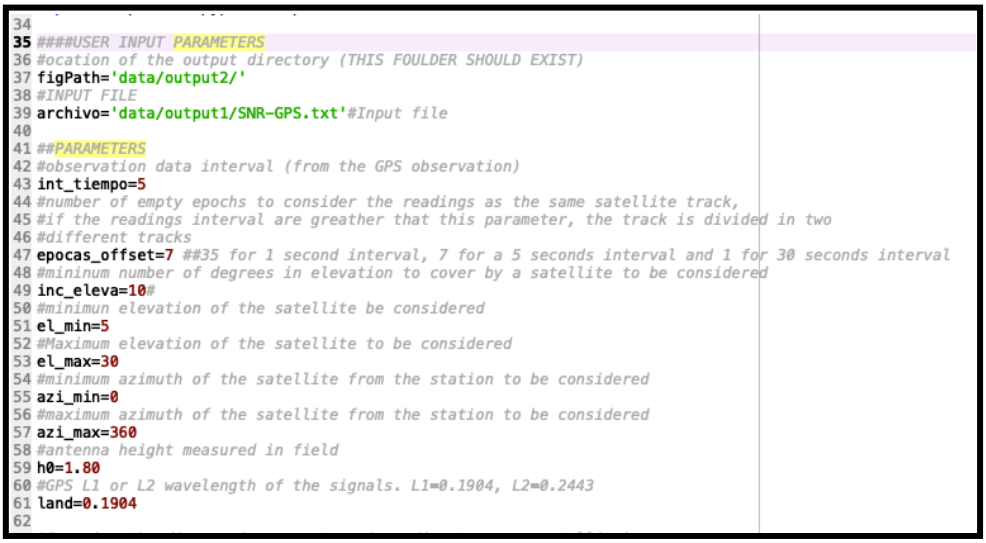
The second step consists in run *SNR-list_satV2.py* software file (this is the second version of the software, at the end of this manual the differences with version 1 are described).

The input for the second step is the output of the first step; it generates an output file per satellite containing the reflection height and the adjusted values for phase and amplitude of the interferometric wave for each individual satellite track. The software also generates a graphical output for each track containing the direct SNR signal, the indirect SNR signal, the computed interferogram and the wave adjusted to the indirect SNR signal.

Before running the software, the user must set some internal input parameters, Figure 6:

1. The time interval between observations: It can be obtained from the observation file header (5 seconds in this example).
2. The minimum and maximum satellite elevations: in case the input file contains values with other elevations. These parameters are set again, by default, to 5 and 30 degrees.

3. The minimum and maximum azimuth of the satellite tracks to be considered: this allows the user to focus on a certain area around the antenna. These parameters are set by default to 0 and 360 degrees.
4. The antenna height: measured from the ground to the antenna reference mark in meters.
5. The working frequency length: 0.1904 m for L1, 0.2443 m for L2.
6. A minimum range of elevation angle to be covered by the satellite to consider the track as valid: set by default to 10 degrees.
7. A minimum number of epochs without observations to assume the observations belong to the same or a new satellite track: In certain cases, some epoch information is lost in the observation file due to signal interruptions or a malfunction, so the software identifies satellite observations that are temporally separated by less than the number of epochs set in this parameter as belonging to the same track. The value depends on the time interval between epochs; it can be set to 1 if the time interval between observation periods is 30 seconds, 7 if the interval is 5 seconds or 35 for 1 second.



```
34
35 ####USER INPUT PARAMETERS
36 #ocation of the output directory (THIS FOULDER SHOULD EXIST)
37 figPath='data/output2/'
38 #INPUT FILE
39 archivo='data/output1/SNR-GPS.txt'#Input file
40
41 ##PARAMETERS
42 #observation data interval (from the GPS observation)
43 int_tiempo=5
44 #number of empty epochs to consider the readings as the same satellite track,
45 #if the readings interval are greather that this parameter, the track is divided in two
46 #different tracks
47 epocas_offset=7 ##35 for 1 second interval, 7 for a 5 seconds interval and 1 for 30 seconds interval
48 #mininum number of degrees in elevation to cover by a satellite to be considered
49 inc_eleva=10#
50 #minimun elevation of the satellite be considered
51 el_min=5
52 #Maximum elevation of the satellite to be considered
53 el_max=30
54 #minimum azimuth of the satellite from the station to be considered
55 azi_min=0
56 #maximum azimuth of the satellite from the station to be considered
57 azi_max=360
58 #antenna height measured in field
59 h0=1.80
60 #GPS L1 or L2 wavelength of the signals. L1=0.1904, L2=0.2443
61 land=0.1904
62
```

Figure 6. Input parameters for *SNR_list_sat.py* software, *h0* is the measured antenna height and *land* is the *L1* frequency.

First, the software creates an output folder structure inside /outpu2/ folder: one folder for each satellite identified by its numerical identifier, creating 32 folders numbered from 1 to 32. If these folders already exist, the program will fail, since they are created

during execution; this prevents previous results from being overwritten or generating more files within the folders with each successive execution.

The software selects each track of every satellite, transforms observed SNR data of each track to linear SNR in Volts/Volts, compute the indirect SNR signal, compute the Lomb-Scargle periodogram from the indirect SNR signal and adjust a wave to the indirect SNR signal by the least-squares algorithm. The maximum reflector height allowed in the output periodogram is set to *2.5* meters, but this value can be changed if the elevation of the antenna is higher, Figure 7.

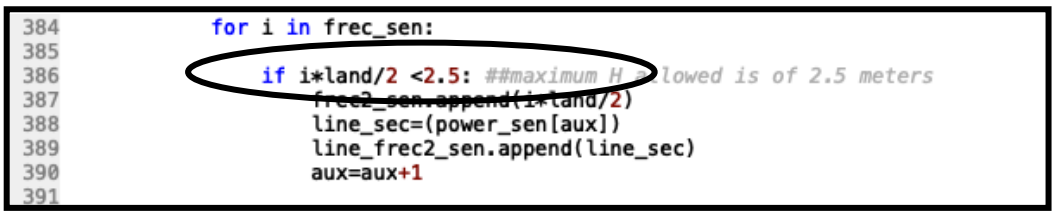


Figure 7. The maximum reflector height allowed in the output periodogram is set to 2.5 meters.

The most important step in this module is to establish conditions for selecting "good" tracks. In this case, based on different experiments, a track is considered valid if: the satellite track contains more than 30 minutes of observation and more than the minimum previously set range angle value of elevation to be covered by the satellite, the power of the dominant frequency is 6 times larger than the media background noise, and the adjustment of the theoretical wave to the indirect SNR signal presents a residual vector with a mean less than 1.3 Volts/Volts and a standard deviation less than 25 Volts/Volts. However, those conditions can be changed by the user (line 436 of the code).

Two output files are generated (*final_values.txt* and *final_values_good.txt* by default), in which each line corresponds with one track and contains the following columns:

1. numerical identifier related to the epochs of the observed file
2. yes/no text field related to the previous conditions (yes indicates "good or valid" tracks)
3. epoch of the observation time
4. year of the observation time
5. month of the observation time
6. day of the observation time
7. satellite identification number
8. initial satellite track azimuth

9. final satellite track azimuth
10. initial satellite track elevation
11. final satellite track elevation
12. adjusted *A* value (including the sign)
13. standard deviation of the adjusted *A* value
14. adjusted ϕ value (in degrees)
15. standard deviation of the adjusted ϕ value (in degrees)
16. computed reflector height
17. satellite-falling/satellite-rising text field

The output file *final_values.txt* contains all the tracks and the output file *final_values_good.txt* only contains the tracks considered as "good or valid". In Figure 8 the results of the *final_values_good.txt* file for satellite GPS-15 can be seen.

For all valid and invalid tracks, a plot is generating containing four figures: the complete signal (SNR), the reflected or indirect signal, the Lomb-Scargle periodogram for the indirect signal, and the reflected signal with the adjusted wave. The figure name includes the numerical identifier, year, month, day, satellite identifier, the initial azimuth and the final azimuth to allow the user to clearly and reliably identify the figure with the corresponding line in the output files.

Results for satellite GPS-15 (*sat_15* folder), are included in the *ReferenceResults* folder so that the user can validate the software operation. Figure 9 shows the track with identifier epoch 88518, 19 January 2019, starting azimuth of 281 degrees and ending azimuth of 300.1 degrees. Specifically, Figure 9a is the SNR data in dB-Hz, Figure 9b is the reflected SNR data in Volts, Figure 9c is the Lomb-Scargle periodogram for the SNR reflected signal and Figure 9d depicts the SNR reflected signal with the adjusted wave in Volts.

final_values_good.txt

```
epoch good_track epoch_time year month day id_sat Azi_ini Azi_end ele_ini ele_end A desvA Phi desvPhi ref_height SatelliteDirection
2369.0 yes 11.788888888888889 2019 1 14 15 281.1 300.1 5.0 30.0 -28.34310421210303 0.9770075146843575 25.118316792722234 0.034468966481176644 1.827 satellite-rising
6207.0 yes 17.230555555555554 2019 1 14 15 170.9 169.0 30.0 5.0 42.05649380106566 1.1997693480220462 212.36172856590933 0.028603368488812085 1.825 satellite-falling
19599.0 yes 11.719444444444445 2019 1 15 15 281.0 300.1 5.0 30.0 -29.27220640754878 1.0287612941674247 22.5219073235819 0.0350767393381446 1.827 satellite-rising
36829.0 yes 11.65 2019 1 16 15 281.0 300.1 5.0 30.0 -28.96685832015718 0.9963580565790704 23.11257400353952 0.03434600884813049 1.827 satellite-rising
40748.0 yes 17.093055555555555 2019 1 16 15 170.9 169.0 29.9 5.0 40.20826540319804 1.2009138761870193 212.3726739861761 0.029947546113098564 1.827 satellite-falling
54058.0 yes 11.579166666666666 2019 1 17 15 281.0 300.1 5.0 30.0 -29.442153322060527 0.9945015343716931 23.91268253251758 0.033718626526290836 1.825 satellite-rising
57978.0 yes 17.023611111111111 2019 1 17 15 170.9 169.0 29.9 5.0 40.581779410307774 1.2373477620546922 212.08207573715322 0.030580840714307751 1.827 satellite-falling
71288.0 yes 11.509722222222223 2019 1 18 15 281.0 300.1 5.0 30.0 -29.80503581048259 1.0006654814358407 24.3140240054571 0.033536600538692552 1.825 satellite-rising
75208.0 yes 16.954166666666666 2019 1 18 15 170.9 169.0 30.0 5.0 41.35001519753724 1.2550780962830043 211.33313539080545 0.030396251415044248 1.825 satellite-falling
88518.0 yes 11.440277777777778 2019 1 19 15 281.0 300.1 5.0 30.0 29.406611569080362 0.9407996598110735 201.39749844966366 0.03190916117709722 1.825 satellite-rising
92438.0 yes 16.884722222222223 2019 1 19 15 170.9 169.0 30.0 5.0 40.56960508642857 1.2058455023967671 211.16976314649492 0.029769171736488787 1.825 satellite-falling
105748.0 yes 11.370833333333334 2019 1 20 15 281.0 300.1 5.0 30.0 -30.48320478424555 0.9870122031801989 24.062531329010312 0.03236966569861607 4 1.826 satellite-rising
109668.0 yes 16.81527777777778 2019 1 20 15 170.8 169.0 30.0 5.0 39.819959336985924 1.1816968699086612 210.42204501122706 0.029711905609403733 1.825 satellite-falling
```

Figure 8. Values of the output file "*final_values_good.txt*" of the second module.

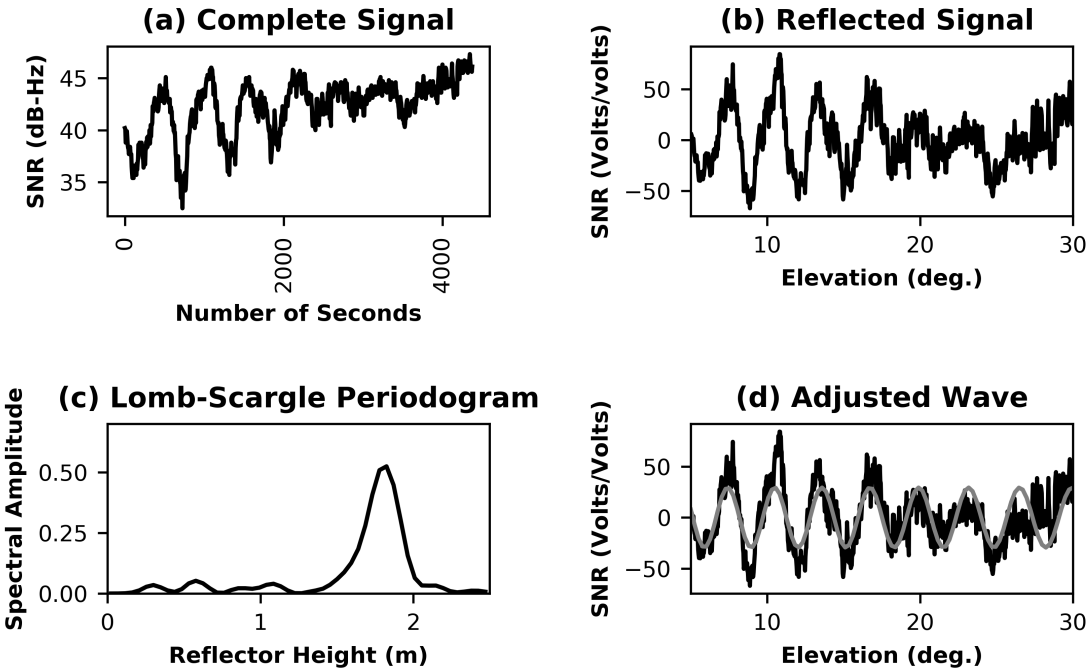


Figure 9. GPS satellite 15 track example results: a) observed SNR data in dB-HZ, b) reflected SNR signal in Volts, c) Lomb-Scargle periodogram for the SNR reflected signal, d) SNR reflected signal with the adjusted wave.

Final step

The final step is to transform variations in the adjusted phase wave value into soil moisture variations (more specifically, volumetric water content *VWC* m^3/m^3).

This step is not included in the software. To do this, in-situ observations are needed as reference data set. Those observations can be obtained from conventional water content reflectometer sensors (Larson et al. 2010, Vey et al. 2016), or from soil data samples (Martin et al. 2019). Considering a linear relationship between GNSS-IR estimated phase variations and reference *VWC* variations, several weeks (or months) of both types of data are necessary to obtain a good relationship. The slope is adjusted using the satellite tracks for which the phase variations presented a stronger linear correlation with in situ soil moisture variations. For example, in Zhang et al. (2017) this correlation is set at 0.9, so only the ascending tracks of GPS satellites 13, 21, 24 and 30 and the descending tracks of GPS satellites 05, 09, 10, 15 and 23 are used. Finally, the slope to be used for all valid tracks should be the mean slope value obtained for the highly correlated satellite tracks.

Final remark about Astropy library

The software works with any version of this library, however, the software considers the first versions of the library where the Lomb-Scargle is include in the stats module, if the

user has a modern version of the library (V3 or V4), this call will produce a deprecation warning alert. In order to eliminate this warning, simply, in the code, activate line number 29 and deactivate line 28, Figure 10.

```
26
27 import os
28 from astropy.stats import LombScargle # to astropy V2
29 #from astropy.timeseries import LombScargle # from astropy V3 or V4
30 import numpy
31 import math
32 from math import pi
33 import matplotlib.pyplot as plt
34
```

Figure 10. Final consideration about astropy library version.

Release notes

The differences between the *SNR_list_sat.py* (version 1 of the software) and *SNR_list_satV2.py* (version 2 of the software are):

1. Introduction of the year in the *final_values.txt* and *final_values_good.txt* output files.
2. Introduction of the adjusted amplitude sign (positive or negative in order to work with the correct phase angle) in the *final_values.txt* and *final_values_good.txt* output files
3. A bug fix that solves the output writing of some satellite tracks that does not corresponds with the satellite under consideration (especially for satellite identification numbers 1, 2, and 3).

References

Larson KM, Braun JJ, Small EE, Zavorotny VU (2010) GPS multipath and its relation to near-surface soil moisture content. IEEE J Selec Top Appl Earth Obs Rem Sens 3(1):91-99. htpps://doi.org/10.1109/JSTARS.2009.2033612

Leick A, Rapoport L, Tatarnikov D (2015) GPS satellite surveying. John Wiley & Sons, fourth edition, 840 pp

Martín A, Ibañez S, Baixauli C, Blanc S, Anquela AB (2019) Multi-constellation interferometric reflectometry with mass-market sensors as a solution for soil moisture monitoring. Hydrol. Earth Syst. Sci. Discuss., https://doi.org/10.5194/hess-2019-560

Sanz J, Juan JM, Hernández-Pajares M (2013) GNSS data processing. Volume I: Fundamentals and Algorithms. European Space Agency Communications, 223 pp

Vey S, Güntner A, Wickert J, Blume T, Ramatschi M. (2016) Long-term soil moisture dynamics derived from GNSS interferometric reflectometry: a case study for

Sutherland, South Africa. GPS Solut 20:641-654. htpps://doi.org/10.1007/s10291-015-0474-0.

Zhang S, Roussel N, Boniface K, Ha M C, Frappart F, Darrozes J, Baup F, Calvet JC (2017) Use of reflected GNSS SNR data to retrieve either soil moisture or vegetation height from a wheat crop. Hydrol Earth Syst Sci 21:4767-4784. https://doi.org/10.5194/hess-21-4767-2017