Nephelometer Level 0 Data Validation

Users Guide for processing script Level0_neph.m

Washington University in St. Louis
St. Louis, Missouri, USA

Prepared by Crystal Weagle
Updated: March 5, 2022

1.0 SCOPE AND APPLICATION

Level0_neph.m was developed to allow SPARTAN team members to broadly assess the usability of nephelometer files obtained from the field. Following user input, the script quickly creates a 6-panel figure that allows the user to visually inspect measured variable that are often linked to degrading nephelometer data quality. This script does not explore optical quantities such as PMT and reference sensor signals, as this is done by neph_diag.m through routine data processing efforts. Note that files may "pass" the level 0 data validation but not pass level 1 validation, or areas within the file may be invalid.

REVISION HISTORY			
Revision No.	Change Description	Date	Authorization

2.0 DIRECTORY STRUCTURES

At the top of the script, under "USER SWITCHES", the user assigns directories as described:

1. direc_input. This is the path to the raw data files that need to undergo level 0 validation. As currently set up, this is a stand-alone folder that is not directly associated with any site. For ease of use, it is recommended that a general raw folder is established to facilitate processing all new incoming data, regardless of the sampling site.
2. direc_output. When files are determined to be either valid or invalid (via user input) they need to be relocated to their site specific "new" or "rejected" folder. Therefore, this directory should point to the main nephelometer file directory that contains subfolders for each site.
3. direc_plot. This is the path to the folder where figures generated through use of this script are saved. Akin to direct_input, this could be a stand-alone folder, with subfolders for each site. Alternatively, this could point to existed site folders where plots are saved. When this directory is called from within the script, the site that the file belongs to is called.

3.0 USING LEVEL0_NEPH.M

All files slated for analysis will be placed in the designated folder in the root directory as described above. The script will loop through all files in the folder.

1. While every effort is made to make sure the various nephelometer files types are readable, if there is an error in reading the file the prompt shown in Figure 1 below. If the user enters "y", the current file that prompted the error will not be read and the script will

move on to the next file. If the user enters “n”, the script will be terminated, and no other files will be read. Entering “n” gives the user a chance to look at the “A” variable and see if they can figure out why the file was not read properly. Common issues are an unexpected date format, unexpected number of data columns, or a random character output by the nephelometer into the file.

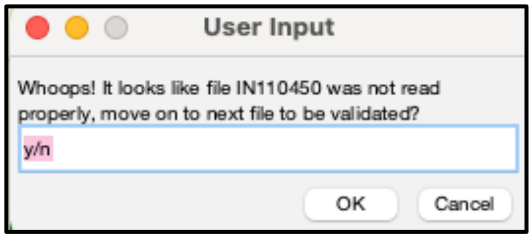

Figure 1. *Screenshot of the User Input prompt that allows the user to determine whether to terminate the script (n) or move on to other files in the folder (y), case sensitive.*

2. After reading in the file, the script estimates the number of days of data contained in the file and displays the prompt shown in Figure 2. The user inputs how many days of data they want shown on each plot. For the file displayed on Figure 2 there are 1.9 days of data. If the user enters “999” all 1.9 days of data will be shown on the plot. If the user enters “1”, the first full day of the data will be shown in the first figure while the remaining 0.9 days will be shown in a second figure. If there were 3.1 days of data and the user selects “3”, then all data will be shown in one figure. The script is designed to distribute any “remainder” data evenly if it is less than 20% of the indicated amount of data.

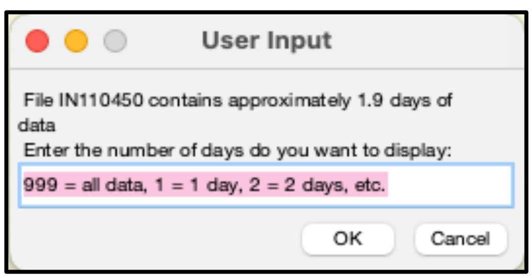

Figure 2. *Screenshot of the User Input prompt that allows the user to determine the number of days' worth of data to display in each figure panel. Only numerical entries accepted.*

3. When the data is plotted, the script will pause to allow the user to interact with the figure panel (e.g., zoom in on suspicious areas). When the user is finished analyzing the plots, press any key to continue the script. This will lead to another prompt that allows the user to decide the following:
    a. Line 1: based on visual inspection of the generated figures, does the file pass level 0 validation? Enter ‘y’ for yes and ‘n’ for no (case sensitive). If there are small sections that are abnormal, it is worth noting but potentially not worth rejecting the file over. The screening that occurs through level 1validation will test for optical issues.

b. Line 2: Enter 'y' if you want to save the generated figure(s), 'n' if not. Figures will be saved in the directory indicated in Section 2.
c. Line 3: Enter the site code for the site that the file belongs to. Since site folders are typically in uppercase, this entry is case sensitive as well.

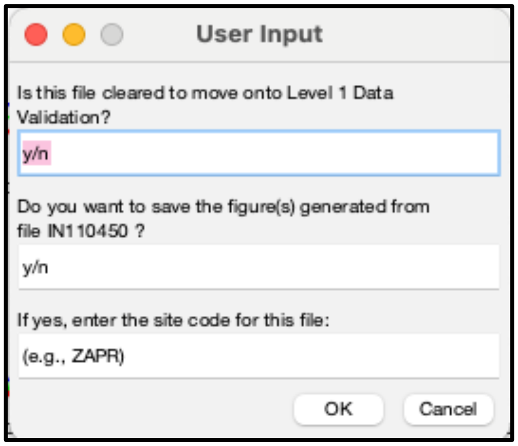

Figure 3. *Screenshot of User Input prompt that allows user to determine whether the file passes level 0 validation, if the generated figures are saved, and to enter the site code that identifies where the data was collected.*

4. After executing the commands following user input in step 3 above, the script will move on to the next file in the 'direc_input' directory.

4.0 INTERPRETING GENERATED FIGURES

Essential to level 0 data validation is the ability of the user to interpret the generated and develop an 'eye" for arising nephelometer issues before they become larger problems with instrument operation. While not exhaustive, below are a few examples to help users gain these skills.

Figure 4 shows the 6-panel figure generated from approximately 3 days of data collected at the site in Halifax, NS Canada (CAHA). Panel 1 and panel 2 shows the total scatter (mM^{-1}) and backscatter (mM^{-1}), respectively. The scatter values are very low at the beginning of the file as the CR system is on at this time as indicated by the lack of displayed flow rate at that time. The scatter values are variable but not extreme, which is characteristic of the CAHA site, but they are not unchanging. Scatter values that show no variation would be a point of concern. Lastly, the general pattern of blue scatter being greater than green which is greater than red is observed. Panel 3 shows the flow rates, which are holding steady at 5.0 LPM and 1.5 LPM at their respective times. Panel 4 shows the fan rpm, the rpm follows the flow rate pattern, and it is clear to see that the fan is working a bit harder to maintain a 5.0 LPM flow rate. Although this file is valid, this may be the first indication that the flow restrictor is accumulating material and needs replacement. The DAC signal shown in panel 5 is also holding a steady pattern inline with the flow rates. Finally, panel 6 is a double y axis plot that shows the temperature and %RH in the file. As expected, temperature and RH show an inverse relationship and when RH begins to climb, the total scatter and backscatter follow a similar pattern as more water is taken up by ambient aerosols.

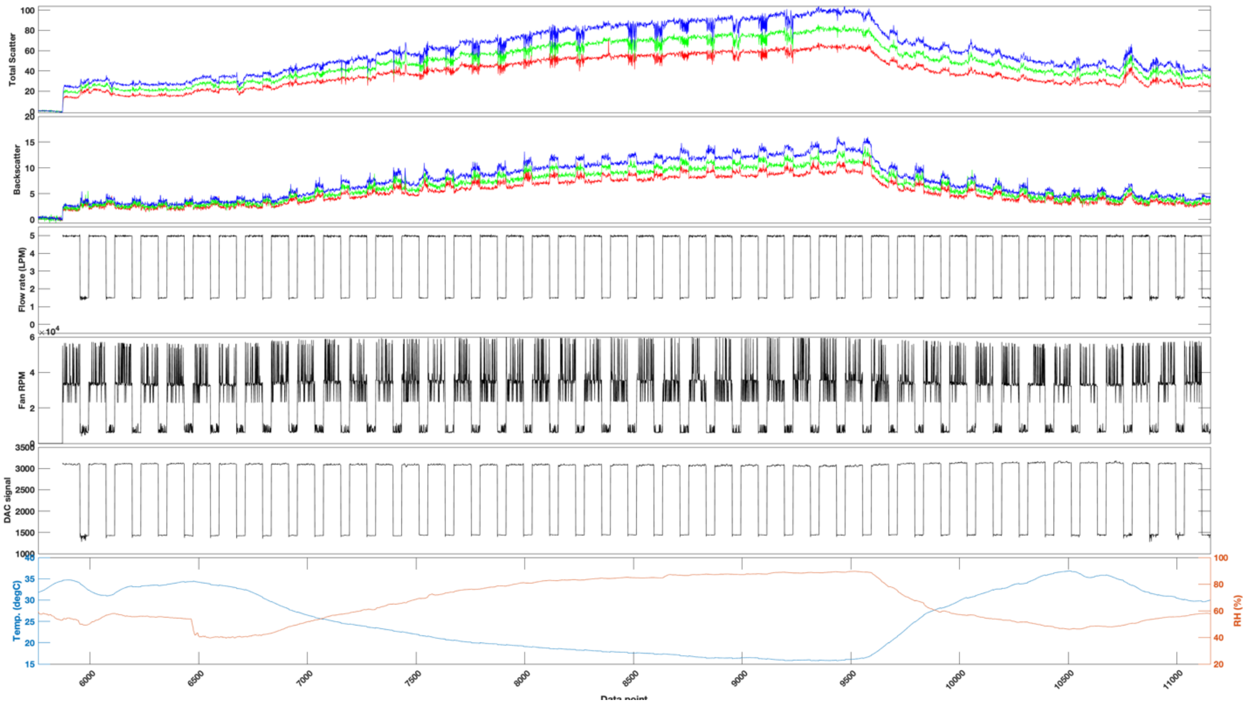

Figure 4. *Six-panel figure generated from a data file from CAHA file IN110450.csv, containing approximately 2 days of data. Data appears to be valid at this stage of analysis.*

Figure 5 shows the figure generated for file IN110471.csv. Panels 2-6 look good; however, panel 1 shows an unexpected pattern of red scatter being higher than green and blue; this is a red flag and it is important to note any optical flags generated from level 1 processing. Red scatter may be higher than other if there is a large, submicron particle population, however red backscatter does not show the same pattern as the total scatter – thus there may an optical issue that is only affecting the forward scatter measurement of the nephelometer.

Figure 6 shows a file from the Mexico City sampling site, with approximately 4 days of data shown. This nephelometer is operating the CR100 system at 3-hour intervals, explaining the periodic drops in scatter, flow rate, fan rpm and increase in DAC signal. Scatter values and ambient meteorological variables (panels 1,2, and 6) are acceptable. However, there are two issues to point out with this file. In panel 3, the flow rate at 5 LPM is achieved readily and appears to be stable throughout the file, however flow rate of 1.5 LPM is not achieved, rather the flow centers around 2 LPM and is highly variable during those periods. This may be an initial indication that the flow restrictor is becoming clogged. The second issue with this file is shown in the fan rpm, shown in the yellow highlight area. Currently, it is not clear what this sustained drop in fan rpm represents, but it is of value to note and determine if perhaps it is another sign if a clogging flow restrictor.

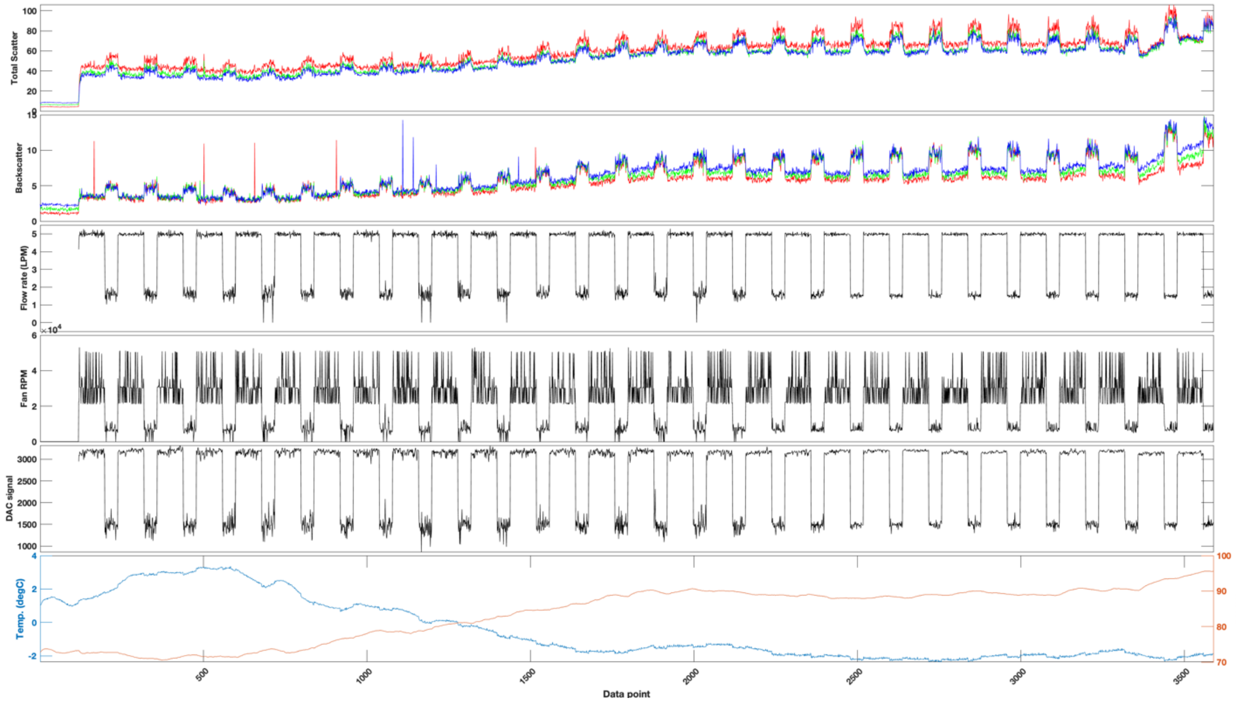

Figure 5. *Six-panel figure generated from a data file from CAHA file IN110450.csv, containing approximately 2 days of data. Data appears to be valid, however the user should expect a possible optical quality flag in the next level of data validation.*

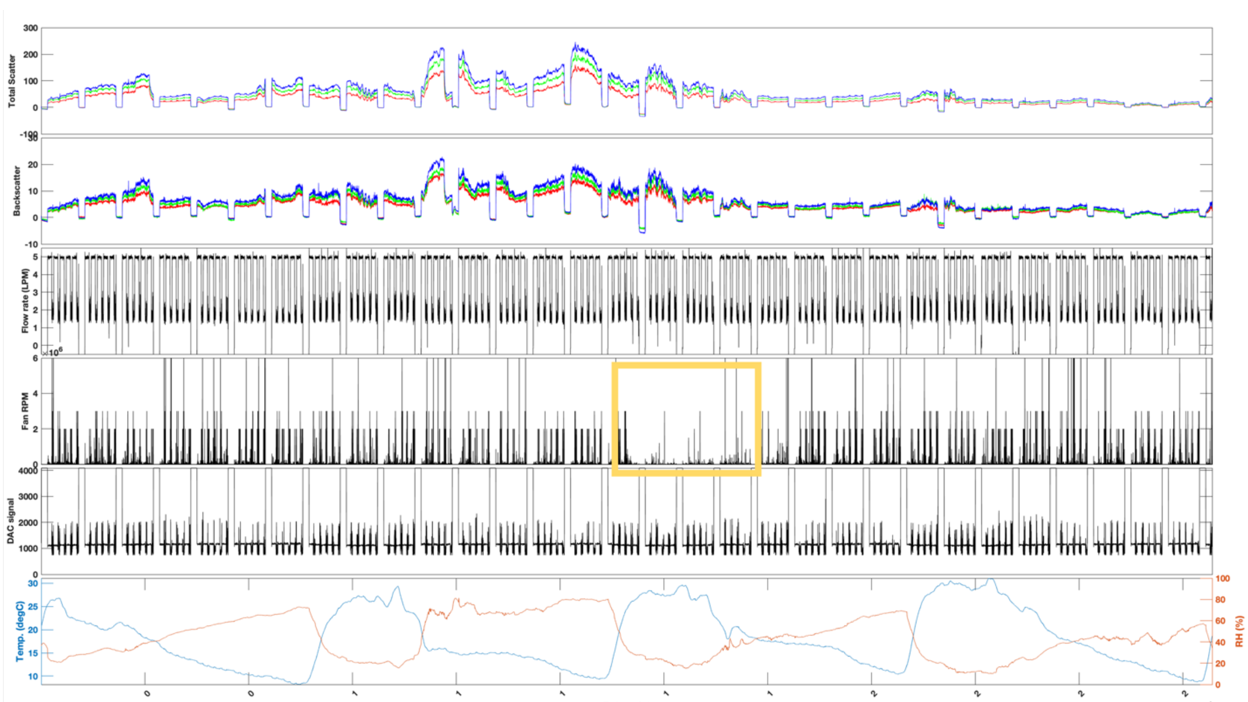

Figure 6. *Six-panel figure generated from a data file from MXMC file IN150204.csv, containing approximately 4 days of data. Data appears to be valid, however there are clear emerging flow rate issues.*

Figure 7 shows a file with some very clear flow rate and fan issues from middle of data file IN520284.csv, collected at the Pretoria, South Africa site (ZAPR). This data is a clear indication of significant fan issues. Flow rate of 5 LPM is not achieved throughout, the DAC signal reaches extreme values as the fan attempts to reach the desired flow rates. While the site manager and site operator may want to try replacing the flow restrictor, too much damage may have been caused.

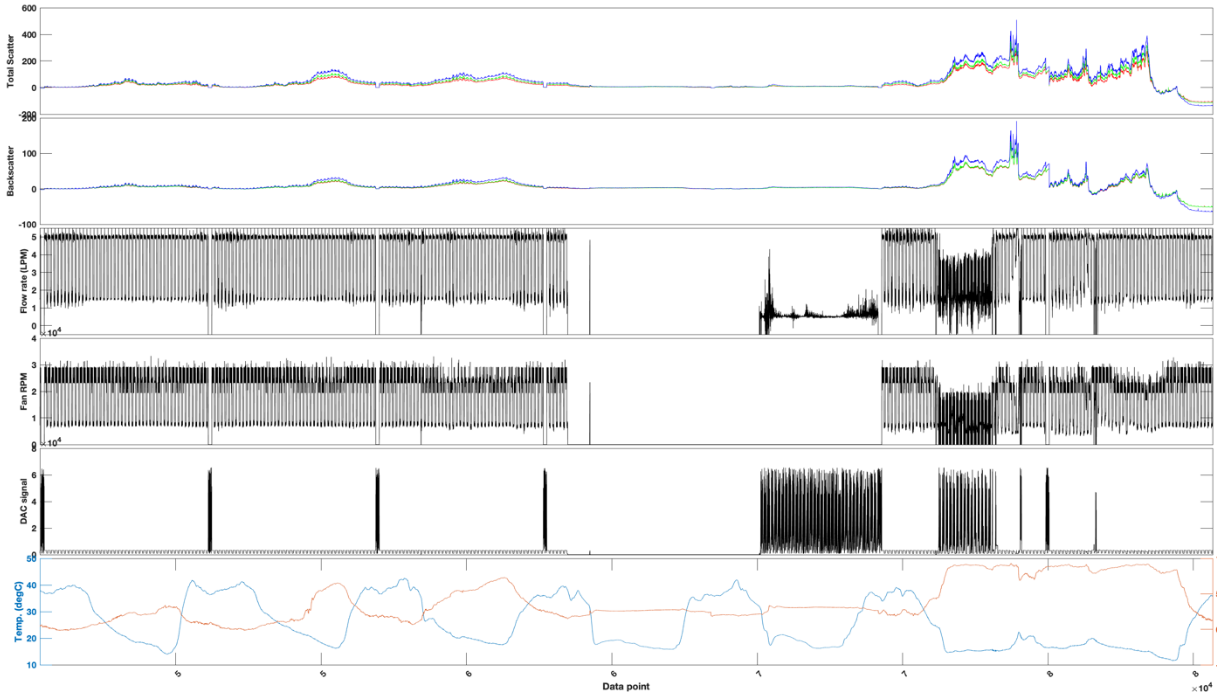

Figure 7. *Six-panel figure generated from a data file from ZAPR file IN5202844.csv, containing approximately 15 days of data. Data is not valid, clear signs of flow rate issues.*