## Preliminary Analysis of TSI Bluesky data with R

## Benjamin de Foy, 9 May 2023

These test programs extract time series data for specific sites and create diagnostic plots. This is not a final product: it is to help you write your own code!

There are plenty of resources to learn R, it would probably be good to work through some introductions.

One thing to keep in mind is that there is “base R” which is the original R. Then there is the “tidyverse” which extends R and does much nicer plots using ggplot2. I am now trying to do as much as possible the tidyverse way, using ggplot2. I would recommend learning this directly, rather than learning how to do plots using base R first. I would work inside Rstudio, using the text editor provided with Rstudio. This will be easier than using just R from the command line.

## Installation:

First install R, then install RStudio on your computer. The instructions and downloads can be found at:

<https://posit.co/download/rstudio-desktop/>

## Directories:

On my Windows laptop, I put my programs in "Dku\_bluesky\_analysis/" on my desktop.

For TSI version 2a from Bangladesh, I put the TSI Bluesky datafiles in "Dku\_cloud\_data/", also on my desktop. The TSI Bluesky datafiles are the 15-min output from the Duke v2 python code that accesses TSILink. Two sample files are provided in the zip file.

On my laptop, the directories are:

"C:/Users/bdefoy/Desktop/Dku\_bluesky\_analysis/"

"C:/Users/bdefoy/Desktop/Dku\_cloud\_data/"

For TSI version2b download: Individual files by site and by time chunk, I put the files in a directory called “Dku\_data\_pull”. For example I have:

C:\Users\bdefoy\Desktop\Dku\_data\_pull\Our Data\Bangladesh\2021-10-01\_2021-11-01\81432123012\Level1.csv

## Execution:

Launch RStudio from the windows menu.

Set the working directory using the menu: “Session -> Set Working Directory -> Choose Directory”: Select “Dku\_bluesky\_analysis” from the file browser.

You can then load a program using: “File -> Open File”: Select, eg. “dku\_read\_bluesky.R”

This will open the file in the editor: you can change options in the code there by editing the text (.R files are text files that can be opened with any text editor).

Click on “Source” to execute the file, or in the console type, eg.: “source(“dku\_read\_bluesky.R”)”

## R Packages:

Rstudio will probably identify the packages you need (eg. library(lubridate)), you can click on install packages to try and install them automatically.

I would recommend installing the packages from the command line, eg. “install.packages(“lubridate”)”. Look at all the “library” commands in the code and make sure you have installed required packages.

If you are missing a package you will get an error about an object not found – it will not tell you that you are missing a package or what that package may be, but you can figure this out by looking at the library statements at the top of the code. You can also google the functions to find the required package.

## Openair R Package:

Many of the plots use the openair package. Refer to the Openair manual for plotting options:

<https://davidcarslaw.com/files/openairmanual.pdf>

## RStudio Notes:

You can execute the code from the Source button or from the console with: “source('dku\_read\_bluesky.R')” or “source('dku\_read\_bluesky.R',echo=TRUE)”

The output file must not be open in Excel when you run the dku\_read\_bluesky.R or you will get the following error: "Error in file(file, ifelse(append, "a", "w")) : cannot open the connection". In this case, close the file and rerun the code.

Once in a while, restart R to clean out your workspace: “Session -> Restart R”

Workspace Data: Under “Tools -> Global Options -> General”: I would uncheck “Restore .RData into workspace at startup” and set “Save workspace to .RData on exit” to “Never”

Any files ending with ~ are emacs backup files. You can safely delete/ignore these.

Csv files are text files that can be read in Excel, but also in any text editor, eg. Notepad. Excel tends to reformat data when reading and writing csv, so sometimes it is safer to use notepad to edit the files. However, you should not have to edit them much – use R to filter the data.

## Data Preprocessing:

### dku\_read\_blueskyv2b.R

This program reads the files in “Our Data” that organized by time chunk and by individual sites.

Set flsites with the inventory to find the serial number from the site name.

Set indir with the directory where you have the data.

Set asites with the sites you want to retrieve, and site\_str with the description to be used in the output file name. This script uses site ID’s not site names (eg. “bgd\_3\_mohakhali”).

You can also set timeminsct if you know when the site was first deployed, so that you screen out when the instrument was being tested in the lab.

### saq\_inventory\_tsdatav2b\_script.R

Read all files in “Our Data” and create an inventory file – Note this could overload your computer, it was not written to be memory efficient.

### dku\_read\_bluesky.R:

This program reads multiple csv files created by Version 2 python scripts from Duke University.

Set “asites” to choose the sites you want (see the inventory file below for site names available). This script uses site names, not site ID’s (eg. “3. Mohakhali”).

The code will create uniform time series, and write them to csv.

The code looks for a scale factor and applies it to the entire time series. The code also creates a vector of unscaled pm25 (pm25\_v0) and writes this out to the 15min file.

There are 2 types of files: \_utc.csv are in GMT time zone, \_lt.csv are in local time. Excel can read the times in the \_lt.csv files.

The code always creates hourly files.

There is an option to output 15min data, which contains both the scaled and the unscaled pm25 time series.

Files with coords in the name contain the lat/lon coordinates of the sites.

## Data Postprocessing

R Code to create plots: in RStudio, open the file in the text editor, then change indir and flroot with the directory and file names containing preprocessed data from dku\_read\_bluesky.R.

### dku\_postprocess.R:

Sample plots of the filtered files created by dku\_read\_bluesky.R.

Check code for options on time selection (timeminsct and timemaxsct), season selection (seasonsct)

When comparing sites, it can be important to filter the data so that you use only times with valid data at all sites. This is done with domatch = 1 in the code.

There is an option to write out csv files of the data actually being plotted (dowritecsv = 1) – this is so that you can check what the code is doing (QA == Quality Assurance).

Check the Openair manual for options, especially for the timevariation plot:

Statistic = ‘mean’ or ‘median’ shows different ranges of data

Key.columns = 4: makes multiple rows of the key if you have so many sites that you cannot see the full key

### dku\_correlogram.R:

Make a correlation plots for multiple sites.

### dku\_boxplot.R:

Compare boxplots for diurnal and weekly timescale.

### dku\_leaflet\_map.R:

Make a map of average pm25 concentrations.

### tbe\_map\_inventory.R:

Make a map of all the stations in the inventory file.

For interactive viewing, open the following file in a web browser: psaq\_bluesky\_dku\_20210715\_20230131\_inv\_map.html

### bdf\_utils.R:

Utility functions, the programs above use "source('bdf\_utils.R')" at the top of the code.

### ebs\_read\_tbe.R:

Program to read TBE csv file into multiple tibbles. Used by tbe\_map\_inventory.R.

### saq\_bluesky\_dku\_20210715\_20230131\_inv\_tbe.csv:

Inventory file of v2a data created by a separate program on my computer: "saq\_inventory\_tsdata.R" (saq is a 3-letter abbreviation for the South Asia Air Quality Project). It contains a list of sites and start/end times as well as scaling factors.

### saq\_bluesky\_bgd\_20211001\_20230430\_inv\_tbe.csv:

Inventory file of v2b data created using saq\_inventory\_tsdatav2b\_script.R.